2012-09

Media Independent Handover for Wireless Full Motion Video Dissemination

Ohleger, Jr., Michael Peter

Monterey, California. Naval Postgraduate School

NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

MEDIA INDEPENDENT HANDOVER FOR WIRELESS FULL MOTION VIDEO DISSEMINATION

by

Michael Peter Ohleger, Jr.

September 2012

Thesis Co-Advisors: Geoffrey G. Xie
John H. Gibson

Approved for public release; distribution is unlimited

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704–0188

1. REPORT DATE *(DD–MM–YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From — To)*
30–9–2012	Master's Thesis	2010-07-05—2012-09-21

4. TITLE AND SUBTITLE

Media Independent Handover for
Wireless Full Motion Video Dissemination

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Michael Peter Ohleger, Jr.

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Naval Postgraduate School
Monterey, CA 93943

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Defense Information Systems Agency

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited

13. SUPPLEMENTARY NOTES

The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government. IRB Protocol number N/A.

14. ABSTRACT

With an increase in the amount of daily UAV flights and the number of Digital Video Broadcast Return Channel Satellite (DVBRCS) suites in the Central Command (CENTCOM) Theater of Operations, the demand for a constant access to the operational picture has also increased. Until recently, there have been limited solutions for enlarging the access to DVBRCS video feeds. With the advent of wireless technologies, such as WiFi, WiMAX, 3G, and LTE, the opportunity to extend the access should be considered. In particular, the IEEE 802.21 standard, known as Media Independent Handover services, could be the solution to not only extending the network beyond the reaches of the forward operating bases, but allowing for no loss in connectivity, due to its ability to conduct seamless handovers, while on the move. In this thesis, a proof of concept evaluation of the compatibility of the IEEE 802.21 standard and the DVBRCS system, using an open source implementation, is presented. This work is to determine if the standard is to be a viable solution for extending the services of DVBRCS to forward deployed units via wireless networks.

15. SUBJECT TERMS

Media Independent Handover, DVBRCS, IEEE 802.21, Mobility

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			19b. TELEPHONE NUMBER *(include area code)*
Unclassified	Unclassified	Unclassified	UU	81	

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited

MEDIA INDEPENDENT HANDOVER FOR WIRELESS FULL MOTION VIDEO DISSEMINATION

Michael Peter Ohleger, Jr.
Major, United States Marine Corps
B.A., Virginia Military Institute, 1996

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN COMPUTER SCIENCE

from the

NAVAL POSTGRADUATE SCHOOL
September 2012

Author: Michael Peter Ohleger, Jr.

Approved by: Geoffrey G. Xie
Thesis Co-Advisor

John H. Gibson
Thesis Co-Advisor

Peter Denning
Chair, Department of Computer Science

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

With an increase in the amount of daily UAV flights and the number of Digital Video Broadcast Return Channel Satellite (DVBRCS) suites in the Central Command (CENTCOM) Theater of Operations, the demand for a constant access to the operational picture has also increased. Until recently, there have been limited solutions for enlarging the access to DVBRCS video feeds. With the advent of wireless technologies, such as WiFi, WiMAX, 3G, and LTE, the opportunity to extend the access should be considered. In particular, the IEEE 802.21 standard, known as Media Independent Handover services, could be the solution to not only extending the network beyond the reaches of the forward operating bases, but allowing for no loss in connectivity, due to its ability to conduct seamless handovers, while on the move. In this thesis, a proof of concept evaluation of the compatibility of the IEEE 802.21 standard and the DVBRCS system, using an open source implementation, is presented. This work is to determine if the standard is to be a viable solution for extending the services of DVBRCS to forward deployed units via wireless networks.

THIS PAGE INTENTIONALLY LEFT BLANK

Table of Contents

List of Figures

List of Acronyms and Abbreviations

3G 3rd Generation
3GPP 3rd Generation Partnership Project
4G 4th Generation
AO Area of Operations
AOR Area of Responsibility
ATA Analog Telephone Adapter
AS-SIP Assured Services Session Initiation Protocol
BS Base Station
CDMA Code Division Multiple Access
CENTCOM United States Central Command
COP Common Operational Picture
CUCM CISCO Unified Communications Managers
DDAN DISN Deployed Access Node
DISN Defense Information Systems Network
DISA Defense Information System Agency
DVB Digital Video Broadcast
DVB-H Digital Video Broadcast – Handheld
DVB-RCS Digital Video Broadcast Return Channel Satellite
EDGE Enhanced Data for GSM Evolution
ESS Extended Service Set
FOB Forward Operating Base
GSM Global System for Mobile Communication
GCCS Global Command and Control System
IEEE Institute of Electrical and Electronic Engineers
IETF Internet Engineering Task Force
LLC Logical Link Control
LTE Long Term Evolution
MAC Medium Access Control
MIH Media Independent Handover
MIHF Media Independent Handover Function
MN Mobile Node
NIC Network Interface Card
NIPRNET Non-Secure Internet Protocol Router Network
PHY Physical layer
PoA Points of Attachment
PoS Points of Service
QoS Quality of Service
SCCP Skinny Call Control Protocol
SIPRNET Secure Internet Protocol Router Network

UMTS Universal Mobile Telecommunications System
UVDS Unified Video Dissemintation System
VoIP Voice over IP
VoSIP Voice over Secure IP
VTC Video Teleconferencing
WLAN Wireless Local Area Network
WRAN Wireless Rural Area Network

Acknowledgements

There are several people whom I would like to thank for their support and encouragement during the past few years. First and foremeost, I would like to thank my wife, Sandi, and my three children, Graham, Jake, and Olivia. Though it was difficult to balance my studies and my family life, they exercised enough patience to allow me to make it through this program with success. While it was difficult at times, their resilience and perseverance kept me on track and proved to be infallible. I love you guys, thank you.

On the professional side of things, I would like to thank my two advisors, Geoff Xie and John Gibson. Thank you for taking on a subject that we knew very little about and helping me turn it into something real. I appreciate the constant encouragement and keeping me on track even when things looked bleak.

I would also like to thank Mr. Bruce Bennett for giving me the opportunity to research a technology that was both cuttting edge and interesting. Ken Quock, Mike DeVries, and Jason Alden, for their assistance and support in all of the DVBRCS related research, testing, and evaluation, their expertise assisted me in meeting the objectives for this project.

Lastly, I would like to thank Subir Das, Antonio de Oliva, and the developers of ODTONE for their patience, their support, and their quick answers to all of the questions that we had regarding 802.21 and ODTONE. Without them, I believe that we would not have been able to get this project to work.

I look forward to any future work in 802.21 and its implementations.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 1:
Introduction

1.1 Motivation

Over the past few years, several commercial communications companies throughout the world have recognized a demand for broader video dissemination technologies over satellite and other types of wireless networks. The Department of Defense (DoD), and more specifically, the Defense Information Systems Agency (DISA), has also taken a keen interest in these technologies. During the most recent military conflicts, such as Operations Iraqi Freedom and Enduring Freedom (OIF and OEF respectively), there has been significant operational need for delivering real-time video services to warfighters on the ground [2].

In 2005, the Tactical Service Provider (TSP) at DISA was asked by Central Command (CENTCOM), to develop a system capable of providing a transportable two-way, IP-based SATCOM system. The result was the Digital Video Broadcast Return Channel Satellite (DVBRCS) system. This system has been widely used throughout CENTCOM as one of the most proficient full-motion video (FMV) dissemination systems available. DVBRCS has the capability to extend full-motion video services across the battlespace, but currently this can be accomplished only through wired technologies available at the individual unit's network level. If the capability to extend this service were available through wireless technologies (802.11, 802.16, etc.), and applications such as the Unified Video Dissemination System (UVDS), now resident within the confines of the Global Command and Control System (GCCS), were made into mobile applications, the benefits could outweigh the cost. Considering the current capabilities, there is nothing available at this time to provide the Common Operational Picture to the warfighter that is about to assault an objective that requires multiple communications relays just to get an idea of what is on the other side of a wall, or where the enemy sniper is located, etc. With wireless extensions for DVBRCS, the ability to see real-time FMV feeds could become a reality.

As Technical Manager of the TSP Joint Capability Technology Demonstration (JCTD), DISA led the integration of a two-way IP-over-SATCOM system extended by a tactical, mobile, WiMAX extension, based on the IEEE 802.16e-2005 standard. The overlying objective of the TSP JCTD was to evaluate the utility of a hybrid communications architecture using standards-based SATCOM and wireless technology solutions to extend global wideband communications

to the tactical edge. The wireless metropolitan area network (WMAN) telecommunications technology (Mobile WiMAX) evaluated through this JCTD could offer significant increases in bandwidth as compared to narrowband tactical wireless solutions deployed today. Mobile WiMAX also addresses many of the shortcomings of WiFi networks, such as data rates, outdoor operation, multipath performance, and the WiFi conflict-based access mechanism that manifests inefficiencies when multiple users are present, particularly as offer-loads increase. Furthermore, Mobile WiMAX offers improvements over traditional 3G technologies, such as a fully packet switched (IP-based) architecture, orthogonal frequency division multiplexing (OFDM), time division duplexing (TDD), multi-level adaptive modulation, stronger error correction techniques, and support for advanced antenna systems. An overview of the key operational functions demonstrated during the TSP JCTD is provided below:

- Provide wireless transport system required to transform every warfighter into an intelligence collector
- Rebroadcast video, imagery, and other broadband services to extend situational awareness and Common Operating Picture (COP) to lower echelons
- Backhaul in-theatre video, imagery, sensor data, and other ISR for forward-deployed tactical users
- Support in-theater tactical collaboration and location-based services

All of the technological advancements and demonstrated operational functions lend themselves well to the case for WiMAX/LTE (Long Term Evolution) as the wireless solution candidate of choice for next-generation tactical wireless systems [2].

One of the solutions offered, by the TSP, to extend these next-generation tactical wireless systems was the IEEE 802.21 standard as a technology enabler. The IEEE 802.21 standard, as defined by the LAN/MAN Standards Committee of the IEEE, is "extensible IEEE 802 media access independent mechanisms that enable[s] the optimization of handover between heterogeneous IEEE 802 networks and facilitates handover between IEEE 802 networks and cellular networks" [1]. This standard provides link-layer intelligence and other related network information to upper layers to optimize handovers between heterogeneous networks. The media types supported includes: Third Generation (3G) Partnership Project (3GPP), 3G Partnership Project 2 (3GPP2), and both wired and wireless media in the IEEE 802 family of standards [1].

IEEE 802.21, also commonly referred to as Media Independent Handover (MIH), features a broad set of properties that meet the requirements of effective heterogeneous handovers. It al-

lows for transparent service continuity during handovers by specifying the mechanisms to gather and distribute information from various link types to a handover decision maker. The collected information comprises timely and consistent notifications about changes in link conditions and available access networks [2].

IEEE 802.21 facilitates multiple methods of handover including both hard-handovers and soft handovers. In hard-handovers, or "break-before-make," there is an abrupt switch between two access points, base stations, or in general, Points of Attachment (PoA); specifically, the current connection is torn down before the next connection is made. Soft handovers, or "make-before-break," require an establishment of a connection with the target PoA while still routing traffic through the serving PoA; thus, for a brief period of time two connections are present (similar to the operation of cellular phones). With soft-handover, mobile nodes remain briefly connected with two PoAs prior to the handoff between heterogeneous networks [5]. Depending on service and application traffic requirements, soft-handovers generally go unnoticed, while some hard-handovers may also go undetected while others require user-intervention. Interactive applications, such as VoIP, are typically the most demanding in terms of handover delays, and high-quality VoIP calls can be served only by soft-handovers. On the other hand, video streaming can accommodate hard-handovers, as long as the vertical break-before-make handover delay does not exceed the application buffer interval delay [2].

DISA is currently looking for solutions to integrate into DVBRCS to provide MIH to the warfighter on the ground. Currently there is no solution available from either Commercial-off-the-shelf (COTS) hardware or from Government-off-the-shelf (GOTS) hardware.

There have been several case studies done by various commercial telecommunications and research companies in Europe regarding implementation of the IEEE 802.21 standard and how MIH will be conducted. While there are currently no known hardware implementations, there has been some developmental software that is becoming available and will be covered in a later chapter, there has been a large amount of conceptualization and a notional framework has been developed by several different research groups and corporations. The notional framework allows users and their applications to state their network access preferences. This framework also allows operators to steer terminal access patterns aiming at maximizing resource usage and increasing user satisfaction. For example, podcasts can be downloaded only when connected to an uncongested WLAN, but web, map/navigation, and e-mail clients can use the cellular network or WLAN access on demand. Currently, this process can only be done manually; users need to

be aware of available access networks and choose to which ones to connect based on very basic information, such as signal strength. If mobile nodes could collect timely and consistent information about the state of all available networks within range and were given a means to control their network connectivity, then a whole range of possibilities would become available [5].

1.2 Research Questions

1. How does IEEE 802.21 provide a seamless transition between current heterogeneous networks in the current IEEE 802 family including but not limited to Local Area Networks (802.3), Wireless Area Networks, or WiFi (802.11), Wireless Metropolitan Networks, or WiMAX (802.16), Personal Area Networks, or Bluetooth (802.15), and Wireless Regional Area Networks (802.22)?
2. Are there other current commercial mechanisms for such handovers which may serve to demonstrate the viability of such a capability?
3. What can the IEEE 802.21 standard provide as a force multiplier to the Department of Defense and will the standard support the requirements for Media Independent Handover (MIH) within the framework designated by DISA as the Tactical Service Provider?
4. What is the best way to allocate hardware that can provide Media Independent Handover (MIH) services in an austere environment, such as Afghanistan, with limited coverage, bandwidth, and resources?
5. Can MIH be integrated into Full-Motion Video (FMV) dissemination systems such as the Digital Video Broadcast Return Channel Satellite (DVBRCS) system?
6. If no commercial system capable of MIH exists, how can the required functionality be emulated to provide a testable proof of concept?
7. Is Internet Protocol version 6 (IPv6) a viable candidate resource for MIH?
8. Is Mobile IPv6 (MIPv6) a viable candidate resource for MIH?
9. Is Session Initiation Protocol (SIP) a viable candidate resource for MIH?
10. Are there any additional protocols that can be considered a viable candidate resource for MIH?

1.3 Thesis Organization

Chapter 2 will provide a sufficiently detailed background to allow the reader to understand the 802.21 technology and related work that has been completed in this field. It will include a detailed explanation of the functionality and current implementations of MIH.

Chapter 3 will present the recommended solution and describe the experimentation necessary to validate that solution. It will include the design aspects of the solution set and the descriptions of products used or developed to implement the solution.

Chapter 4 will provide an assessment of the results of the experimentation performed, as well as any products used or developed to support the solution or demonstration effort.

Chapter 5 will provide concise conclusion statements and present recommendations for the implementation of the proposed solution, given that a solution is possible. Finally, it will provide areas for further study, to include general problem statements for those areas.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 2:
Background

This chapter identifies the current state of IEEE 802.21 Media Independent Handover (MIH) services and the technology that is purported to be supported by this standard. It further addresses the problem presented by DISA in terms of their needs and requirements for an MIH solution. Finally, this chapter discusses related work within the area of the IEEE 802.21 standard.

2.1 The IEEE 802.21 Standard

The IEEE approved the 802.21 standard in 2009, and since then development of a working software solution has primarily been accomplished through commercial vendors, like Telcordia (now Applied Communications Sciences), or research groups where open source versions of Media Independent Handover software, such as ODTONE, have been developed. In this section, we will provide a generalized overview of the 802.21 standard and what ODTONE provides.

Figure 2.1 shows an overview of MIH framework as defined by the IEEE 802.21 standard [1]. The framework consists of three primary services: Media Independent Event Service (MIES), Media Independent Command Service (MICS), and Media Independent Information Service (MIIS). MIES may indicate or predict changes in a state and transmission behavior of the physical and link layers. Common MIES provided through MIH function (MIHF) are "Link Up," "Link Down," "Link Parameter Change," and "Link Going Down." MICS enables higher layers to configure, control, and obtain information from the lower layers including the physical and link layers. The information provided by MICS is dynamic information comprised of link parameters, while information provided by MIIS is made up of static parameters. MIIS provides a unified framework for obtaining neighboring network information that exists within a geographical area. It assists the higher layer mobility protocol in acquiring a global view of available heterogeneous networks to conduct effective seamless handover. The standard defines information structures called information elements (IEs) for MIIS. These IEs are classified into two groups: access network specific information and PoA specific information [6].

A handover event occurs when a Mobile Node (MN), connected to a particular network, moves out of the range of that network and connects to a different network. This is common in cellular networks, where mobile phones frequently move out of the coverage area of one cell tower and

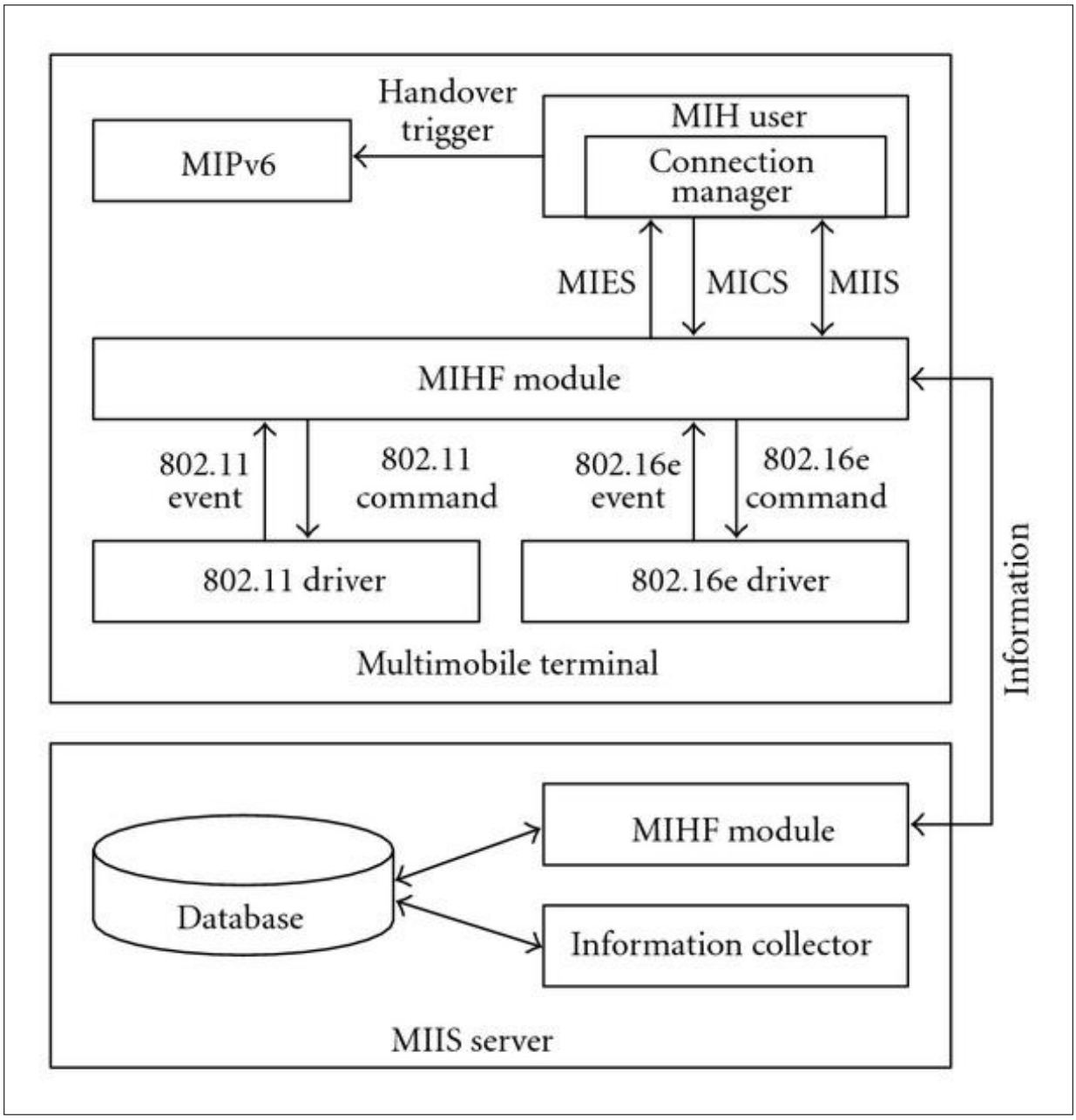

Figure 2.1: MIH framework as defined by the IEEE 802.21 standard. From [1].

into that of a neighboring tower. There are two types of handover, as illustrated in Fig. 2.2. The standard handover process would interrupt all the user sessions (e.g., VoIP calls, FTP file transfer) currently in progress at the MN. Seamless handover differs in that the MN maintains the original user sessions while attempting to connect to a new network, and once the new network connection is up, the handover action automatically migrates the user sessions to the

new connection. This seamless handover allows for total continuity of network communication with only a minor increase of message latency during the handover process.

Seamless handover cannot be efficiently performed without close coordination between the MN and the networks involved. The 802.21 standard designed a new function to control access to layer 2 of the network model, and the function provides a new service access point to allow for information queried by the upper layers. In order for this to work, both mobile devices and the network must implement the standard. This key function, known as the Media Independent Handover Function or MIHF, is located between the layer 2 wireless technologies and IP at layer 3 [7]. The primary role of the MIHF is to provide asynchronous and synchronous services through well defined Service Access Points (SAPs) for link layers and MIH users. In a system containing heterogeneous network interfaces of IEEE 802 types and cellular types (such as 3G, 3GPP, 4G, etc.), the MIHF helps the MIH users to implement effective procedures to couple services across these heterogeneous network interfaces. MIH users utilize services provided by the MIHF across different entities to query resources that are required for a handover operation between heterogeneous networks [1].

Each SAP consists of a set of service primitives that specify the interactions between the service user and provider. In the MIHF specification, two types of SAPs are defined, media independent and media dependent. The media independent SAPs allow the MIHF to provide services to the upper layers of the mobility management protocol stack, the network management plane, and the data bearer plane. Upper layers need to subscribe with the MIHF as users to receive MIHF generated events and also for link-layer events that originate at layers below the MIHF but are passed on to MIH users through the MIHF. MIH users directly send commands to the local MIHF using the service primitives of the MIH_SAP. Communication between two MIHFs relies on MIH protocol messages. Media dependent SAPs allow the MIHF to use services from the lower layers of the mobility management protocol stack and their management planes. All inputs from the lower layers of the mobility-management protocol stack into the MIHF are provided through existing media-specific SAPs such as medium access control (MAC) SAPs, physical layer (PHY) SAPs, and logical link control (LLC) SAPs.

The three primary services that make up the MIHF are the Media Independent Event Service (MIES), the Media Independent Command Service (MICS), and the Media Independent Information Service (MIIS). These services are all managed and configured through service management primitives as outlined in section 6.2 of the approved standard. This configuration is done

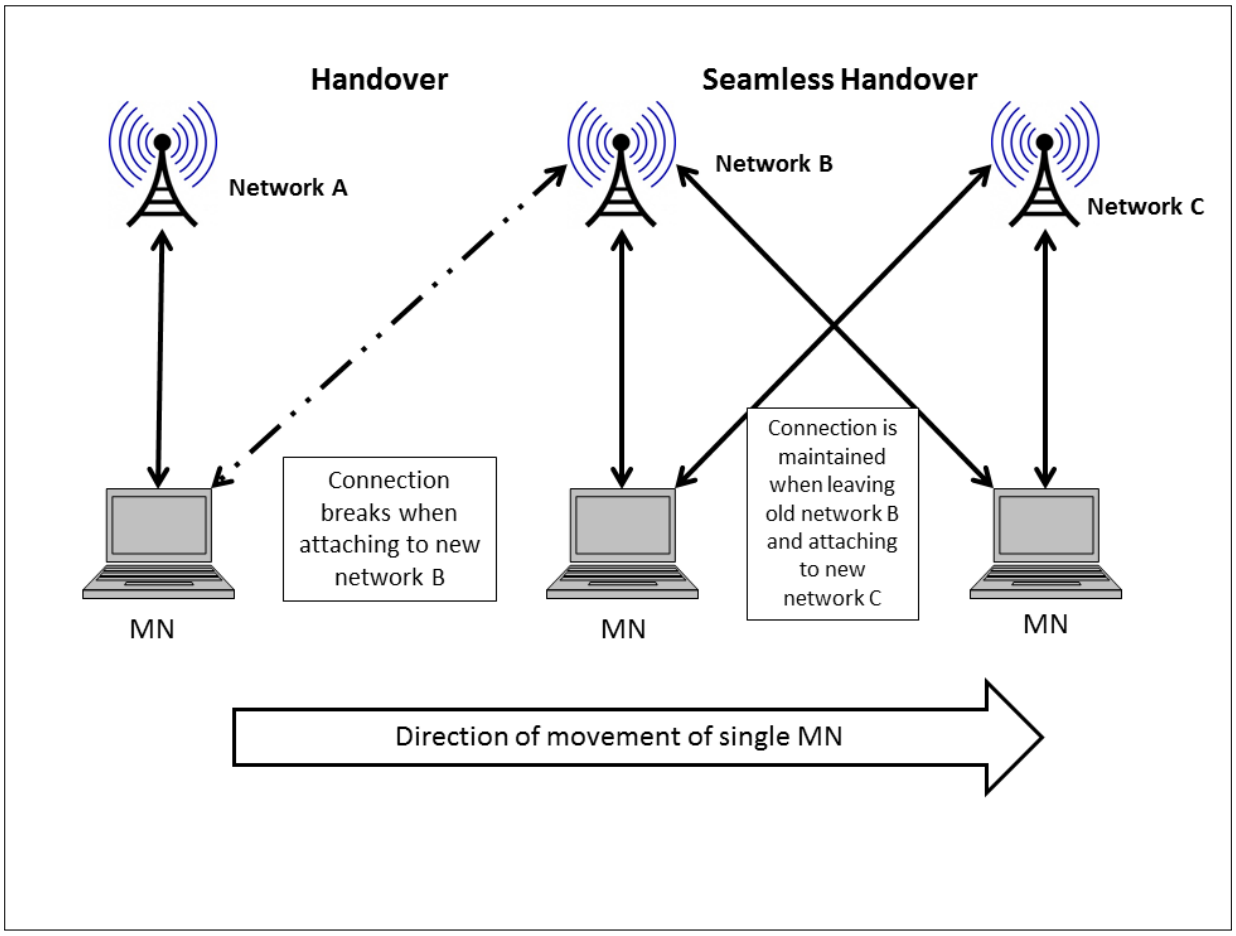

Figure 2.2: Handover and seamless handover being conducted by a single MN on the move. Note that the MN's connection is broken when leaving an old network before attachment to a new network when a regular handover is being conducted.

through the following service management functions [1]:

- MIH capability discovery
- MIH registration
- MIH event subscription

To identify the services supported by a peer MIHF, the MIH node performs an MIH capability discovery. If a legitimate capability is found to be in existence it will register with, subscribe to, and provide network communication with the one that it deems to be the best one with which to connect.

The MIES is used to indicate changes in state and transmission behaviors of the physical, data

link, and logical link layers or predict any changes in the state of these layers. Events originate from the MIHF or in Layer 2 or below of the protocol stack, with the destination of an event being the MIHF or Layer 3 or above. The eventual recipient of the event is located within the node that originated the event or in a remote node. In the situation where the event is local, messages propagate from the Layer 2 and below (PHY, MAC) to the MIHF and from the MIHF to Layer 3 and above. If the event is a remote event, the messages will propagate from the MIHF in one protocol stack to the MIHF in the peer protocol stack [1].

The MICS enables Layer 3 and above to control the physical, data link, and logical link layers. Layer 3 and above will also control the reconfiguration or selection of an appropriate link through a set of handover commands. If the command service is supported by the MIHF, any command coming from the MICS will force the MIHF to abide by the request, which the MIHF must execute. Commands within the MICS are originated by the MIH user (usually MIH commands) or by the MIHF itself (usually Link Commands). The destination of a command is to the MIHF or to any lower layer, the recipient of which is located either in the protocol stack that originated the command or within a remote protocol stack. Commands will be client or network initiated, and handovers within the MICS will be vertical [1]. The information provided by MICS is dynamic information composed of link parameters such as signal strength and link speed; whereas, information provided by the MIIS is static in nature and is composed of parameters such as network operators and types of service and cost [1]. There are a number of commands identified in the standard to allow MIH users to configure, control, and retrieve information from the lower layers including, but not limited to MAC, Radio Resource Management, and PHY. Per the standard, Figure 2.3 shows link commands (Link_Capability_Discover, Link_Event_Subscribe, Link_Get_Parameters, etc.) and MIH commands (MIH_Link_Get_Parameters, MIH_Link_Configure_Thresholds, etc.), while Figure 2.4 shows remote MIH commands sent by MIH users to the MIHF in a peer protocol stack. A remote MIH command delivered to a peer MIHF is executed by the lower layers under the peer MIHF as a link command; or is executed by the peer MIHF itself as an MIH command; or is executed by an MIH user of the peer MIHF in response to the corresponding indication [1].

The following two figures provide a list of both Link commands (Figure 2.5) and MIH commands (Figure 2.6), per the 802.21 standard.

The MIIS, provides all of the information necessary for the MIHF to discover and obtain net-

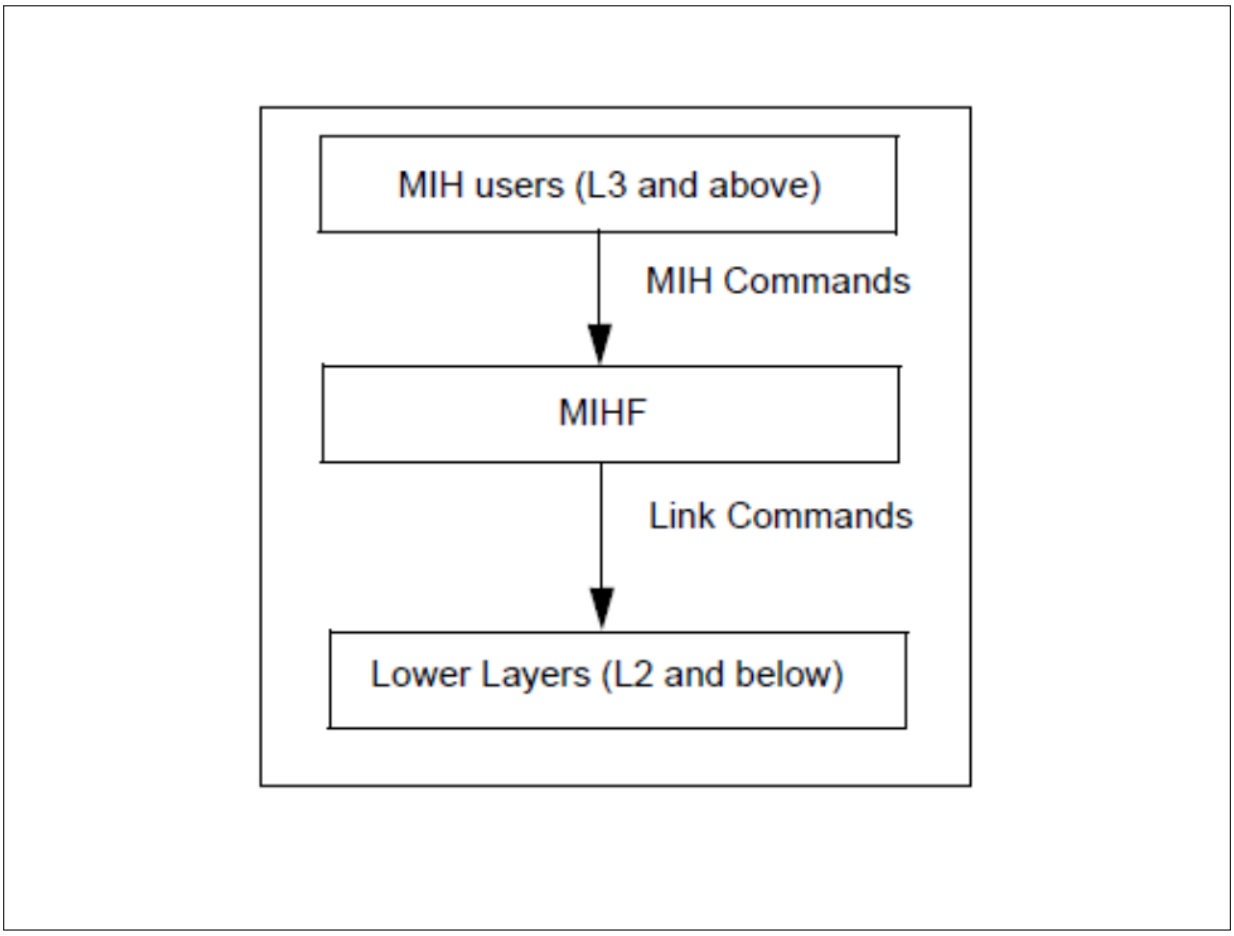

Figure 2.3: Link commands and MIH commands. From [1].

work information within a geographical location to facilitate handovers. Any neighboring network information can also be discovered and obtained within this same framework to provide optimal network selection and access. MIIS primarily provides a set of information elements, the information structure and representation, and a query/response mechanism for information transfer. Information elements are classified into three groups [1]:

1. General Information and Access Network Specific Information: general overview of the different networks providing coverage within an area.
2. POA specific Information: information about different PoAs, such as base stations or access points, for each of the available access networks.
3. Other information, such as proprietary network type, quality of service or Internet Service Provider (ISP) information that is access network specific, service specific, or vender or

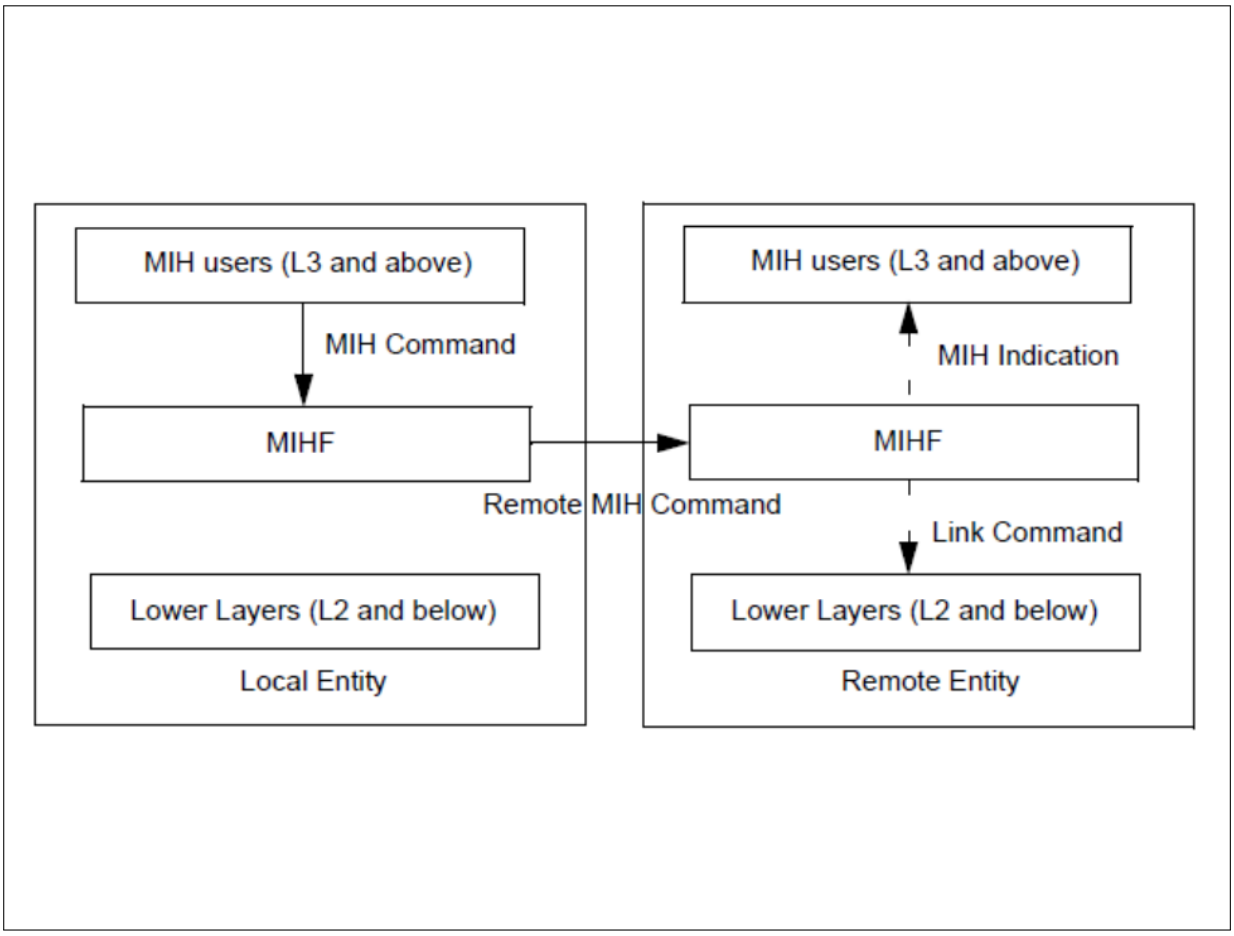

Figure 2.4: Remote MIH Commands. From [1].

network specific.

The information can also include inter-technology handover policies. Lastly, the MIIS also supports a push mode, where the information can be pushed to the MN by the operator, as required. Information is made available to the MIIS via both lower and higher layers, providing the ability to access information about all networks in a geographical area from any single L2 network, depending on how the IEEE MIIS service is implemented. The MIIS will either rely on existing access media specific transports and security mechanisms or L3 transport and L3 security mechanisms to provide access to the information [1]. The MIIS will typically provide static link-layer parameters such as channel information, the MAC address and security information of a point of attachment (PoA) [1]. In a recent conversation with the chairman of the IEEE 802.21 working group, Dr. Subir Das, we confirmed that the MIIS is required for

Link command	Comments	Defined in
Link_Capability_Discover	Query and discover the list of supported link-layer events and link-layer commands.	7.3.9
Link_Event_Subscribe	Subscribe to one or more events from a link.	7.3.10
Link_Event_Unsubscribe	Unsubscribe from a set of link-layer events.	7.3.11
Link_Get_Parameters	Get parameters measured by the active link, such as signal-to-noise ratio (SNR), BER, received signal strength indication (RSSI).	7.3.12
Link_Configure_Thresholds	Configure thresholds for Link Parameters Report event.	7.3.13
Link_Action	Request an action on a link-layer connection.	7.3.14

Figure 2.5: Link commands. From [1].

network discovery, however if there is no need to discover neighboring network information, this service can be skipped.

2.2 Commercial and Open Source Development

The only commercial development that we knew of prior to this study was an MIH Server component designed by Applied Communication Sciences (ACS), and an MIH client node designed by InterDigital. These products were developed and tested in 2008, and were not available for our use in this study due to licensing and legal constraints. Since there was no commercially available product that we could use for the testing phase of this project, we chose to use the open source software known as ODTONE. This software provides the SAPs, the MIHF, the MIIS, the MIES, the MICS, an MIH User, and several other internals such as a DHCP server and client, ICMP SAP, a DNS user, and a dummy server all for testing purposes. It was in-

MIH command	(L) ocal, (R) emote	Comments	Defined in
MIH_Link_Get_Parameters	L, R	Get the status of a link.	7.4.14
MIH_Link_Configure_Thresholds	L, R	Configure link parameter thresholds.	7.4.15
MIH_Link_Actions	L, R	Control the behavior of a set of links.	7.4.16
MIH_Net_HO_Candidate_Query	R	Network initiates handover and sends a list of suggested networks and associated points of attachment.	7.4.17
MIH_MN_HO_Candidate_Query	R	Command used by MN to query and obtain handover related information about possible candidate networks.	7.4.18
MIH_N2N_HO_Query_Resources	R	This command is sent by the serving MIHF entity to the target MIHF entity to allow for resource query.	7.4.19
MIH_MN_HO_Commit	R	Command used by MN to notify the serving network of the decided target network information.	7.4.20
MIH_Net_HO_Commit	R	Command used by the network to notify the MN of the decided target network information.	7.4.21
MIH_N2N_HO_Commit	R	Command used by a serving network to inform a target network that an MN is about to move toward that network, initiate context transfer (if applicable), and perform handover preparation.	7.4.22
MIH_MN_HO_Complete	R	Notification from MIHF of the MN to the target or source MIHF indicating the status of handover completion.	7.4.23
MIH_N2N_HO_Complete	R	Notification from either source or target MIHF to the other (i.e., peer) MIHF indicating the status of the handover completion.	7.4.24

Figure 2.6: MIH commands. From [1].

tended for the ODTONE software to not only provide an actual framework for MIH, but also to be a self-contained testing platform for handover [1]. One group out of Portugal, has implemented and tested network-assisted handovers using the ODTONE software, with much success. Their testbed consisted of three Points of Attachment (PoAs), with co-located Points of Service (PoSs), two of which were equipped with an IEEE 802.11a wireless network interface and one with an IEEE 802.11g wireless network interface; an MIIS server, and a mobile node supporting IEEE 802.11 technologies [8].

In one test they were looking for the "MIH Link Going Down" indication messages, which is delivered when a Layer 2 connection is expected (predicted) to go down (Link Down) within a certain time interval. This event can be an indication to initiate handover procedures. The MIHF will receive this event from the Link Layer, and then will pass this notification onto the MIH

user that has subscribed for this notification [1]. The MIH user takes different actions on receipt of this notification, and then prepares for handover. These same messages were noted during our testing via Wireshark, when we toggled the wireless radio off and on, while connected to an 802.3 connection.

Most of the general testing of ODTONE, if not all of it, has focused on lab testing and we speculate that our testing was one of the first attempts to test 802.21 in a field environment. Further, this is the first testing scenario that has been attempted using DoD equipment.

2.3 Implications in DoD FMV Systems

DVBRCS has grown in capacity from 16 suites that were originally installed in the CENTCOM theater of operations back in 2006, to well over 180 suites currently operating in the same theater. DVBRCS originally communicated with the commercially owned EUTELSAT W6 satellite that operated over the CENTCOM area. As recent as 2009, in a partnership between the DoD and the Australian Department of Defense, the Wideband Global SATCOM System (WGS) 2 constellation was launched. This satellite was positioned over the Indian Ocean for use by CENTCOM in Afghanistan, Iraq, and other parts of Southeast Asia. After the launching of this satellite, DVBRCS ended the terms of their contracts with EUTELSAT, and moved to a single transponder on WGS-2. This move, allowed DVBRCS to increase its bandwidth, thus allowing for an increase in the amount of video services that are allowed to be broadcast by, and received, on a single terminal. The growth in DVBRCS has proven to be a greatly successful one, for it has become one of the most widely known and used dissemination tool in CENTCOM. The popularity of DVBRCS has created a need for further dissemination of video via the Unified Video Dissemination System (UVDS), which is part of the Global Command and Control System being used by CENTCOM and, provides a user the capability to search archived video.

With the introduction of many of these new capabilities, and the need to reach outside the boundaries of existing forward operating bases (FOBs) with existing networks, a requirement has been identified to extend these lower coverage, tactical wireless networks in such a way that the distribution of real-time information to nodes that are outside the boundaries of these wireless networks becomes a reality.

The IEEE 802.21 standard may offer a solution to extend network connectivity beyond the boundaries of the FOBs. This can be accomplished by extending the existing WiFi networks

within the FOB (as was done at Camp Leatherneck, Afghanistan, in 2010) to a mobile WiMAX node, additional mobile WiFi hotspots, or by employing organic or host nation cellular networks and infrastructure, using the 802.21 standard.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 3:
Concept

This chapter considers practical applications of the IEEE 802.21 standard, and how we utilize it to meet DISA's requirements. Here, testing scenarios of three sorts are presented. The first are demonstrations designed as "canned" experiments to show how MIH work in ODTONE, the second is a lab environment designed to showcase the functionality of the components of the MIHF, and the third is a field testing environment where full motion video is broadcast while seamless handover events are triggered to force the components of the MIHF to react and perform a handover as prescribed in the standard. In all three testing scenarios, we use open source software called ODTONE for experimentation and testing of the IEEE 802.21 standard.

3.1 Initial Considerations

In the planning phase of our testing, we identified several testing objectives to be considered by our work. Primarily, we wanted to perform a proof of concept validation that would provide us with a platform to move forward into further testing. We identified our initial requirements and came up with the following questions:

1. Will ODTONE work within the framework of DVBRCS, and if so, what are the possible limitations that we might face when attempting to perform handovers while streaming video?
2. What is the optimal protocol for conducting handovers (Mobile IP, SIP, or Host Identity Protocol)?
3. For what trigger events should we be looking when attempting to conduct handovers from an heterogeneous network (such as the 802 family of networks) to a mobile network (such as 3G, 4G, or LTE cellular)?
4. Should we begin to consider possible handover mechanisms for CDMA, GSM, or Iridium?

We envisioned an over all architecture wherein 802.21 is integrated with the DVBRCS. After speaking with Dr. Subir Das, Chairman of the IEEE 802.21 Working Group, we developed the architecture as depicted in Figure 3.1. This architecture is designed to accomodate an MIHF (complete with a LINK_SAP) at the DVBRCS hub, one at the suite, and one at the MN.

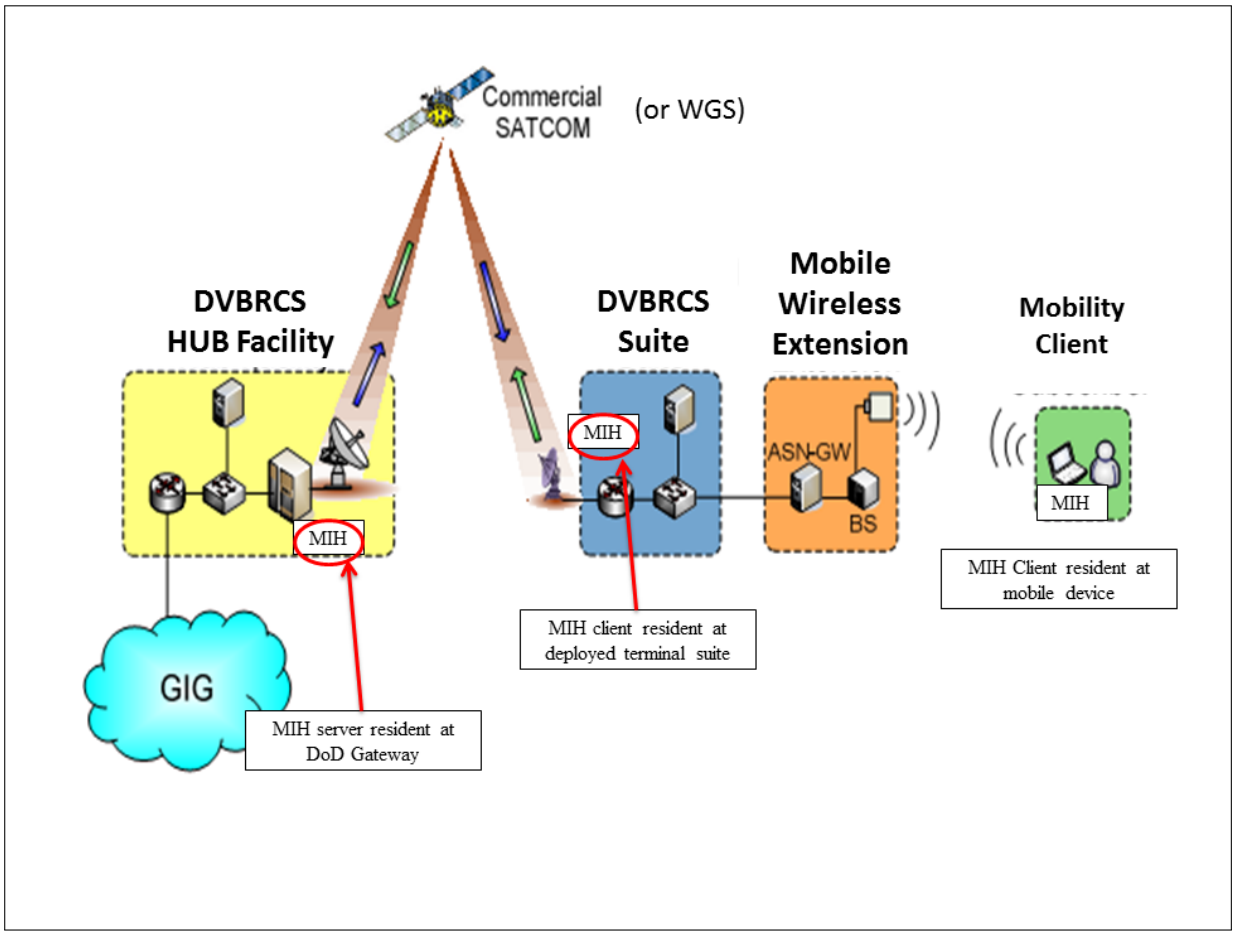

Figure 3.1: Notional DVBRCS architecture with MIHF locations at both the DVBRCS hub, suites, and client. After [2].

3.1.1 Experimental Study of SIP

As part of our initial considerations, we conducted an experimental study to get a better understanding of the Session Initiation Protocol (SIP). The following section provides the information that we gathered about SIP as we prepared to move ahead with our research into IEEE 802.21.

We took a hard look at which protocol would be best suited for conducting handovers between MNs and which best suited the needs and requirements of DISA. We originally considered using MIPv6, as it had more applicability to mobile use. However, this approach presented problems that will be discussed in more detail in a later chapter.

Arrangements were made with DISA to obtain PCAP traces of lab generated traffic, in simulations, and actual exercise generated traffic going through the DVBRCS system located in

Maryland. Analysis of these packets was originally intended to observe network behavior and identify protocols used in the DVBRCS network sessions.

Upon further review of DISA's requirements, and in keeping with the requirements for implementing the 802.21 standard, we decided to review what was available at the DISA gateway sites in Landstuhl, Germany and Lago, Italy. Understanding that SIP was an option for use as a handover, albeit slower than MIPv6, we decided to look at how VoIP calls were handled at the teleport sites to assess if SIP was a viable option for use, since there would be a requirement to have an MIHF present at the DVBRCS hub. DISA's goal was to incorporate 802.21 into DVBRCS, and more specifically into the Defense Information Systems Network (DISN) Deployed Access Node (DDAN). DDAN is a suite of systems that is an offshoot of DVBRCS and has the following functional components: VoIP, VTC, NIPRNet, SIPRNet, and VoSIP.

SIP, as defined in RFC 3261, is an application-level control protocol for setting up, modifying, and terminating real-time sessions between participants over an IP data network [9]. SIP is a signaling protocol adopted by the IETF as an open standard for VoIP and Video Teleconferencing (VTC) services. We found out, through packet traces provided by DISA, that the DDAN and the DISA gateways only use the Skinny Call Control Protocol (SCCP) as the signaling protocol for VoIP telephony. SCCP is a proprietary CISCO protocol that provides VoIP functionality to analog telephones. While it is suitable for use over digital VoIP phones, it is a legacy system that has historically been used between CISCO Unified Communications Managers (CUCM) and Analog Telephone Adapters (ATAs). An additional protocol, the ITU's Packet-based Multimedia Communications Systems, known as H.323 [10], is also used in conjunction with SCCP for the purposes of providing VTC capabilities to the same systems. Using these two protocols together to provide functionality carries a lot of overhead as it spends a large amount of effort in the setup and connection of the phones themselves before any communication between users has taken place.

However, once we determined that we needed to isolate SIP packets as a solution for MIH services, the scope of the analysis changed slightly. Ideally, we would have seen some SIP packets coming through the hub, if DISA had provided traces that included VoIP or VTC usage; but instead DISA provided UDP traces for video feeds, which did not contain any of the packets in which we were interested. DISA was then able to capture packets from several VoIP phone calls. This traffic was from calls made to an individual phone from a CUCM, and vice versa; we ran into another issue regarding these traces that will be discussed later in this paper. These calls

provided plenty of useful information but did not include any traces that used SIP as a signaling protocol. Instead, as mentioned before, the call managers that DISA uses at the Gateway operate using SCCP as their signaling protocol.

Based on this information, we made the decision to compare SCCP with SIP, running our own experiment to investigate the viability of replacing SCCP with SIP as a signaling protocol at the DISA Gateways, thus providing a means for conducting handovers. This would allow for the extension of a VoIP call through the DDAN, outside of the normal range of the services provided by 802.3 and 802.11 networks inside a Forward Operating Base, to units with HN cellular phone infrastructure or mobile WiMAX.

The SIP experiment was conducted using two X-Lite 4 Softphone clients and the OfficeSIP server, both freeware available for use on Windows machines. A softphone client is a VoIP phone that is used on a desktop or laptop computer for VoIP and VTC calls; it also has instant messaging capabilities. The OfficeSIP server software allows you to set up a SIP server on a laptop or desktop to provide SIP services over an ad-hoc wireless or intra-network. Our system was set up using one laptop computer, running a softphone client and the OfficeSIP server, and a desktop computer, running the other softphone client, connected over an ad-hoc wireless network. Once OfficeSIP was configured, we were able to make several phone calls from one machine to another using the softphones. These phone calls included VTC, voice, and IM transmissions, which were identified in the collected PCAP traces as Real Time Protocol (RTP) traffic. Following these collections, we compared the SIP traces with the SCCP traces only to discover that the SCCP traces did not have any RTP traffic included in the traces. Therefore, any RTP traffic being sent via SCCP was implied in accordance with the protocol design.

Figure 3.2 shows the typical handshake observed between the first softphone, OfficeSIP server, and the second softphone. The first few packets are used to initiate a SIP call, with the Session Description Protocol (SDP) defining the session type. At the time of the first invite, the server provides a response to the softphone that initiated the call with a "STATUS 100: TRYING" message letting the initiating softphone know that the call is being attempted. Once the second phone receives the invite it will send a "STATUS 180: RINGING" message back to the server to let the first phone know that the second phone is ringing. Once the connection is made, a "STATUS 200: OK" message will be sent from phone 1 to phone 2 via the server, followed by an acknowledgement of the same status message. Once this handshake has been made, a SIP acknowledgment will be sent and the RTP stream will follow. It is important to point out that the

RTP stream is the vehicle by which any real-time traffic is passed across the SIP connection; in the case of this experiment the real-time traffic was VoIP, VTC, and IM. During our ODTONE testing and evaluation, we made every attempt to replicate this same type of network setup. In a real-world scenario, SIP would support streaming video being transmitted and disseminated across the DVBRCS and DDAN suites, thus making SIP a viable candidate for handover with respect to DISA's requirements.

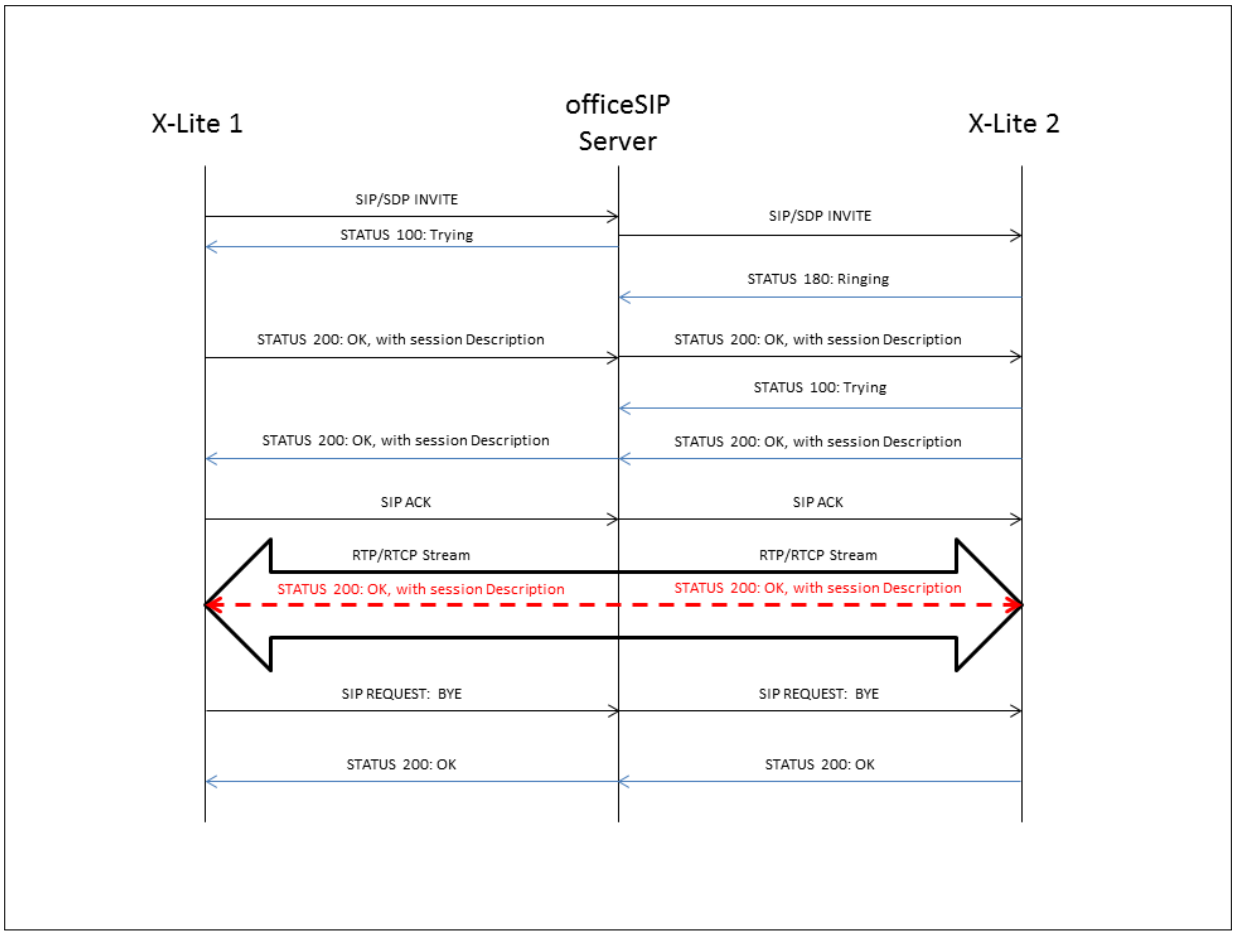

Figure 3.2: Experimental network showing the handshake between two X-Lite softphone clients and the OfficeSIP server

For our SIP trace, we collected over 50,000 packets in the experimental architecture. Of these packets, we found 477 sent SIP packets, of which 274 packets were resent due to dropped calls and other connectivity problems (busy signals, terminated requests, and other client errors). We also found a total of 25 successful RTP sessions that included VoIP, VTC, and Instant Message transmissions. The average setup time for each SIP session was 1769 ms, with the minimum

setup time being 4 ms, and the max being 10822 ms. The max SIP setup time can be attributed to connectivity problems within the ad-hoc wireless network as it was set up for the experiment. These problems arose from an unreliable connection that we had generated within the ad-hoc network. While we had only two end devices attached to a single ad-hoc wireless AP, the signel kept dropping during the experiment. Once we were able to bring the connection back on-line between the two machines, we were able to continue the experiment.

In the DISA provided traces, we observed several phone calls between the CUCM and an individual phone. The handshake, as seen in Figure 1, that took place between the CUCM and the phones did not have any significant differences from what would be observed as normal behavior. It was observed, however, that there were about 23 messages between the two before any media would have been transmitted. One of the packets that were observed that raised some questions was a TCP Window update. The TCP window update is simply used for receive window flow control. If the receive buffer is full and larger packets of information must be passed, a TCP window update request is sent to increase the size of the window for the transmission.

In the SCCP trace, there were no RTP packets collected. Therefore, it was difficult to make a direct comparison between SCCP and SIP for purposes of comparing the full handshake, average setup time, and calculating errors. However, we were able to make comparisons regarding setup time based on the time stamp of the individual packets. In the trace that was collected, the amount of time that it took before the receipt of the StartMediaTransmissionMessage was a little over 5 seconds. This was mostly due to the amount of messages between the CUCM and the phone and the length of time it takes to setup the phone so that it is able to make and complete a valid call. This relative latency in the protocol can be attributed to these setup messages required in order to power the phone and initiate the transmission between the end device and the CUCM.

The functional differences between the two protocols are shown in Figure 3.4. There seems to be a direct mapping between the setup procedures of the SCCP trace and that of the SIP trace. While it only took eight steps before RTP packets were observed in the SIP trace, it took approximately 23 steps before we would have seen the RTP packets in the SCCP trace, again this is due to the setup procedure of the phone by CUCM. While we were unable to gather any RTP from the SCCP call, we were able to produce a statistical comparison between SIP and SCCP, as shown in Figure 3.5. One of the key takeaways from this graph is the statistic on time. All of the traces take place over a very similar time period of about thirteen minutes

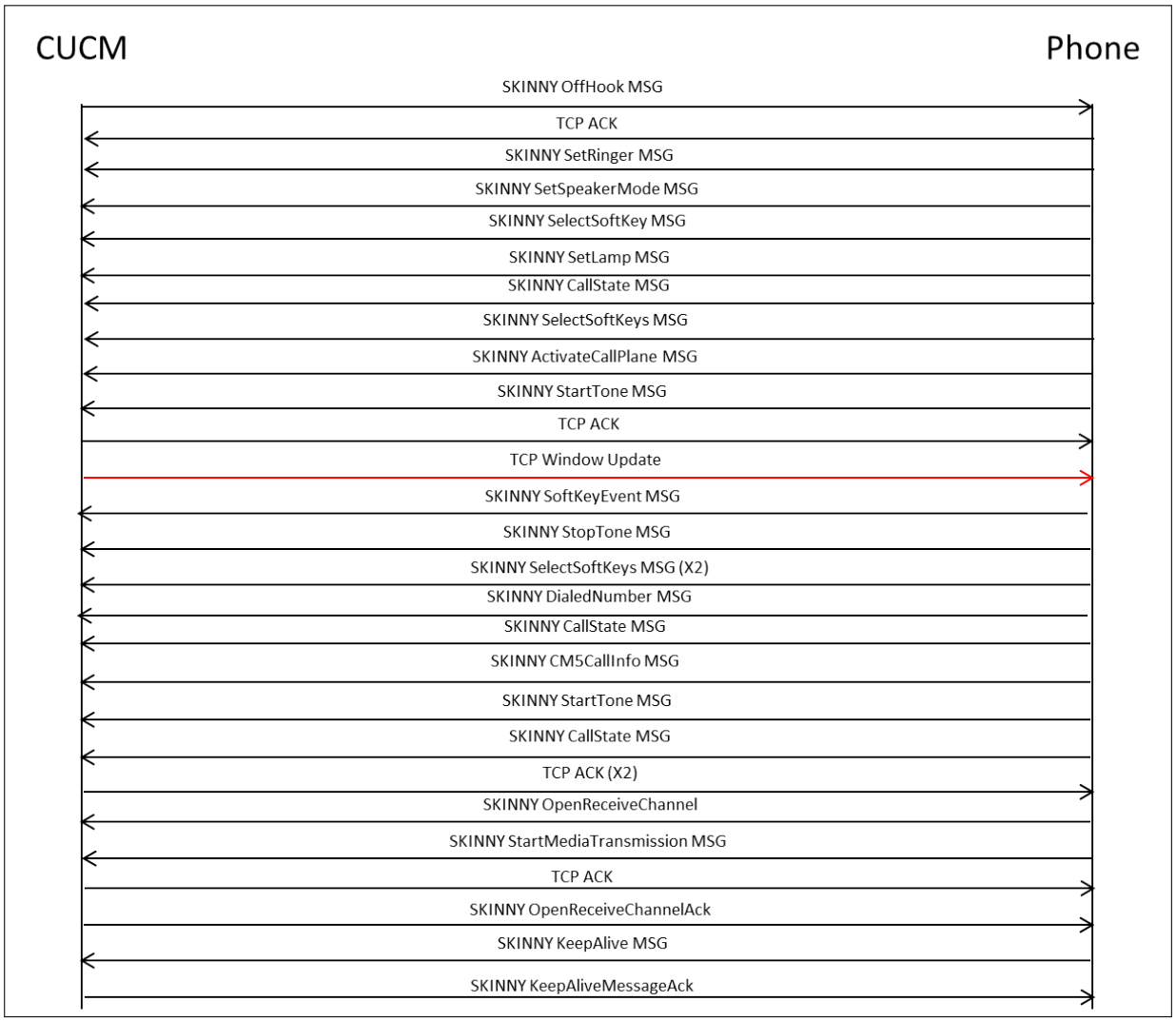

Figure 3.3: SCCP handshake observed in one of the DISA provided traces. This trace is a from the DISN Gateway in Landstuhl, Germany.

each. The longest period of time noted in the SCCP trace is that of the time between the keep alive messages; as noted before, there should be RTP packets incorporated in this portion of the trace, so it is observed that there is a keep alive message sent approximately every 30 seconds throughout the duration of the transmission despite the fact that there are no RTP packets being sent.

In the SIP trace, the “STATUS 200: OK” message exchange, which act as a keep alive message, occurs approximately once every 1500 packets, rather than sending the status message based on a time stamp. In some cases, we found multiple instances of this message being sent, only to

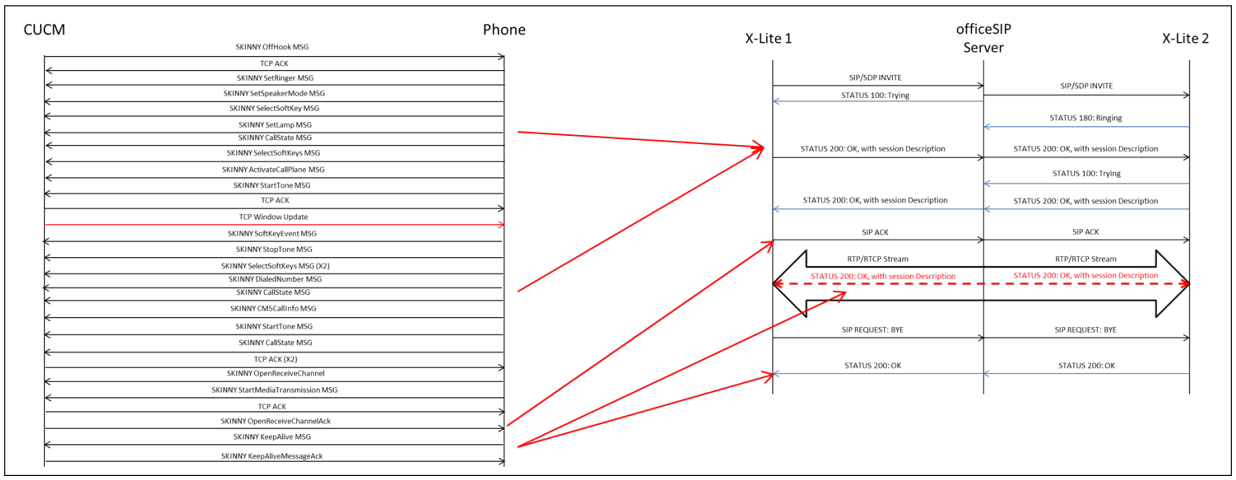

Figure 3.4: Functional comparison between SIP and SCCP.

	SIP	SCCP (CUCM -> Phone) w/o RTP	SCCP (Phone -> CUCM) w/o RTP
Length (bytes)	8011895	45434	91338
Elapsed Time	13:32	12:15	13:09
Packets	50079	463	936
Time Between 1st & Last Packet	812.698	728.816	784.862
Avg. Packets/Sec.	61.621	62.339	1.193
Avg. Packet Size	143.985	81.521	81.324
Bytes	7210607	37744	76120
Avg. byte/sec.	8872.432	51.788	96.985
Avg. Mbit/sec	0.071	.0004	.0007

Figure 3.5: Statistical analysis of the two protocols. Note SCCP is broken down into two columns, one set of traces was from the standpoint of CUCM to phone, and the other is from phone to CUCM.

discover that this was being done due to failing links in the network. Once we were able to establish a solid connection between the two SIP clients, the status message was sent periodically during the RTP stream to ensure that the link was still valid.

Of the functional comparison, we were able to derive a correlation between the two protocols in terms of setup and transmission of media5. Since SIP does not have setup procedures for end items, all of the setup functions that the CUCM does for the phone can be limited to the SIP/SDP invite. This appears to provide the same functionality and does it in a smaller amount of steps, thus providing better efficiency overall.

One of the key features of note between the functional comparisons is the KeepAliveAckMessage (CUCM to phone), KeepAliveMessage (phone to CUCM) in SCCP and the STATUS 200: OK message in SIP. These two messages, as described above, serve as a "heartbeat" between the end devices used in each of the traces, ensuring that the path between the two phones (in SIP) or between the CUCM and the phone (in SCCP) is still a valid link.

This study was a stepping stone to understanding how SIP will work within the context of the IEEE 802.21 standard and how DISA might best support Media Independent Handover services within the framework of the DVBRCS. Future analysis, testing, and experimentation will be conducted to ensure that SIP is indeed a valid protocol for these types of services.

The information gathered in this study has provided a valid argument that SIP will work as a replacement for SCCP at the DISA Call Managers. There are only a few limiting factors in the replacement process: those being equipment and AS-SIP. The equipment issue that DISA faces in its endeavor to replace SCCP with SIP is that some of the current Call Managers will not upgrade to SIP and therefore, must be replaced by equipment that will allow for upgrade.

There is another possible solution into which we still have to look, that being a middleware solution between SCCP and SIP. However, there was no available information that could be found regarding a "bridge" mechanism that would allow SCCP to work in conjunction with SIP for the purposes of providing the necessary protocol for possible MIH services to be valid.

It was announced at the DISA Customer Conference early in 2011 that DISA will be phasing out SCCP at their Gateway Call Managers and will be implementing a new proprietary version of SIP known as Assured Services Session Initiation Protocol, which constitutes all of the same functionality of the standard version of SIP, but with secure services added to the protocol. AS-SIP was tested in house by DISA in 2010, but has not been tested within the framework noted here.

Unfortunately, we were unable to use both OfficeSIP and the X-Lite softphones in our ODTONE experiments, as all of our ODTONE experimentation was done with Ubuntu and OfficeSIP and

the X-Lite softphone clients are only available for use in a Windows environment, which lead us to the Kamailio [11] SIP server and the Twinkle [12] SIP softphone for use in our ODTONE experiments.

3.2 ODTONE Demonstrations

ODTONE is open source software designed to fulfill the mechanism required for MIH to occur. In essence, ODTONE supplies the implementation of a MIHF, supporting its services (MIES, MIIS, MICS), as well as supporting mechanisms (capability discovery, MIHF registration, Event registration, etc.) [3], which facilitates the communication between L2 and L3 to allow handovers to occur. In order to run ODTONE, which will operate in either a Linux environment or in a Windows environment, you must install the Boost libraries to compile all of the components of ODTONE. Since the executable code is all in the C programming language, you must use gcc in Linux or Microsfot Visual Studio (successfully tested with version 11) in Windows. This includes, but is not limited to, the MIHF, the LINK_SAP, the MIH_USR, and the MIIS_RDF_SERVER. To conduct a working demonstration of ODTONE on a live network, you must use the SAP_80211 LINK_SAP, which is a working 802.11 LINK_SAP, and is different from the LINK_SAP used in the demo, as the LINK_SAP in the demo and is designed to only trigger events between the MIH_USR and the MIHF, thereby serving simply as a means to assess the event trigger handler call mechanisms.

The first testing scenario we attempted involved a local demo developed in conjunction with ODTONE designed to demonstrate the functionality of the MIHF, a LINK_SAP, and an MIH_USR all on one machine. In order to conduct this test, the MIHF must be configured with the MAC address for the wireless NIC on the laptop and the IP address for the local machine must be included. Then the MIH_USRmust be configured so that it can talk to the MIHF locally. Once these steps were complete, ODTONE is started in the following order:

1. Start the MIHF
2. Start the local LINK_SAP (there is a requirement to have one LINK_SAP for every MIHF)
3. Start the MIH_USR

The design for this demonstration can be seen in Figure 3.6. The purpose of this demo was to show how events are generated by the LINK_SAP when the MIH_USR connects to the SAP and communicates with the MIHF.

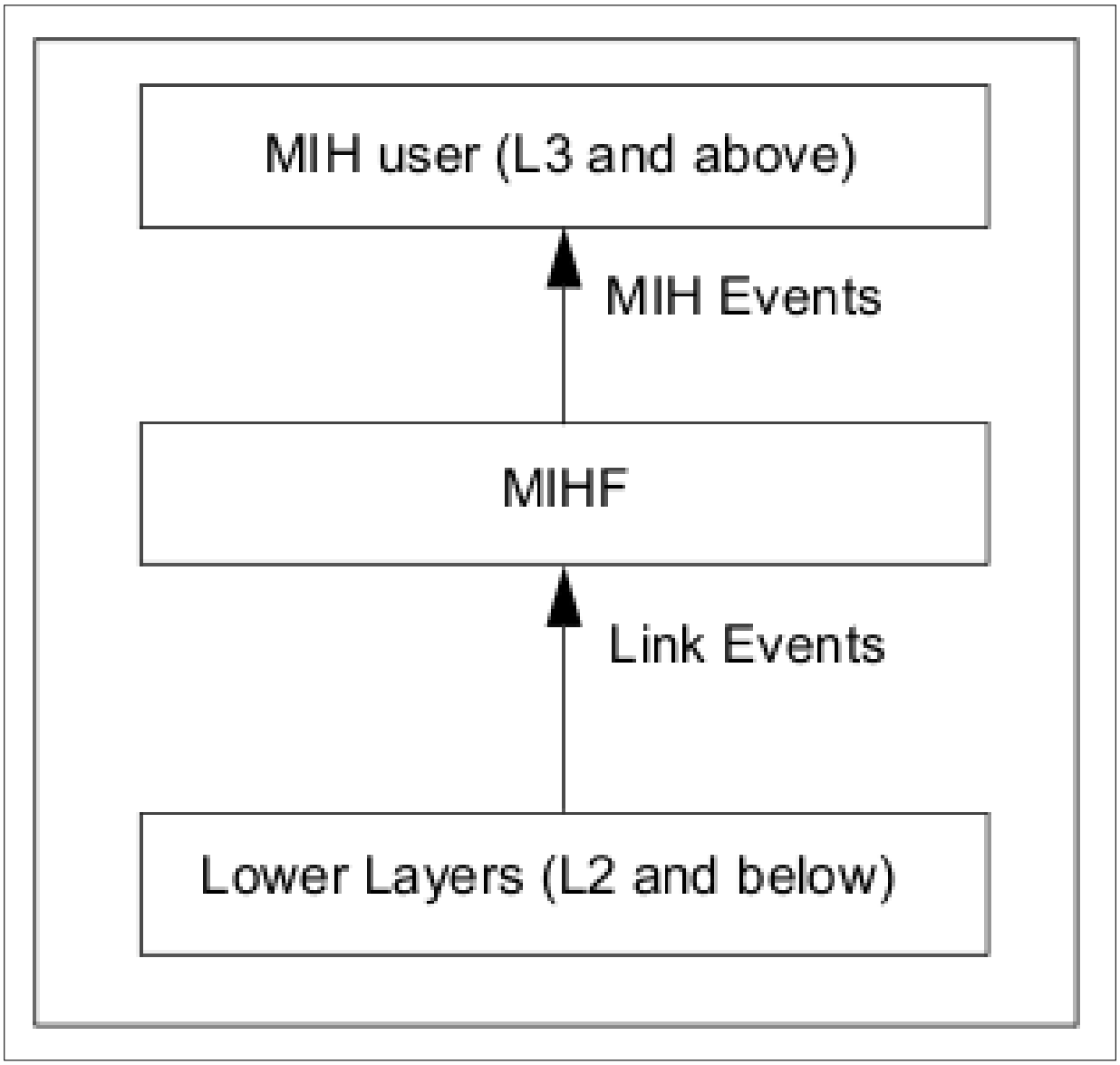

Figure 3.6: Local demo architecture on a single machine designed to show events generated by the LINK_SAP. From [3].

The next scenario we conducted was a remote demo developed by ODTONE to demonstrate the ability of a MIH_USR to obtain event notifications that happen on a LINK_SAP located on another machine. The preponderance of our initial testing was done with this demo. For this demo, two computers are required. The first machine hosts an MIHF (MIHF1) and the MIH_USR, while the second machine hosts an MIHF (MIHF2) and a LINK_SAP. Once the configurations were complete, the componenets are started in the following order:

1. Start MIHF1 on laptop 1.
2. Start LINK_SAP on laptop 1.
3. Start MIHF2 on laptop 2.
4. Start the MIH_USR on laptop 2.

Figure 3.7 depicts the layout for this demo. Event notifications for both the local and remote demonstrations are propagated from the LINK_SAP to the MIH_USR [3].

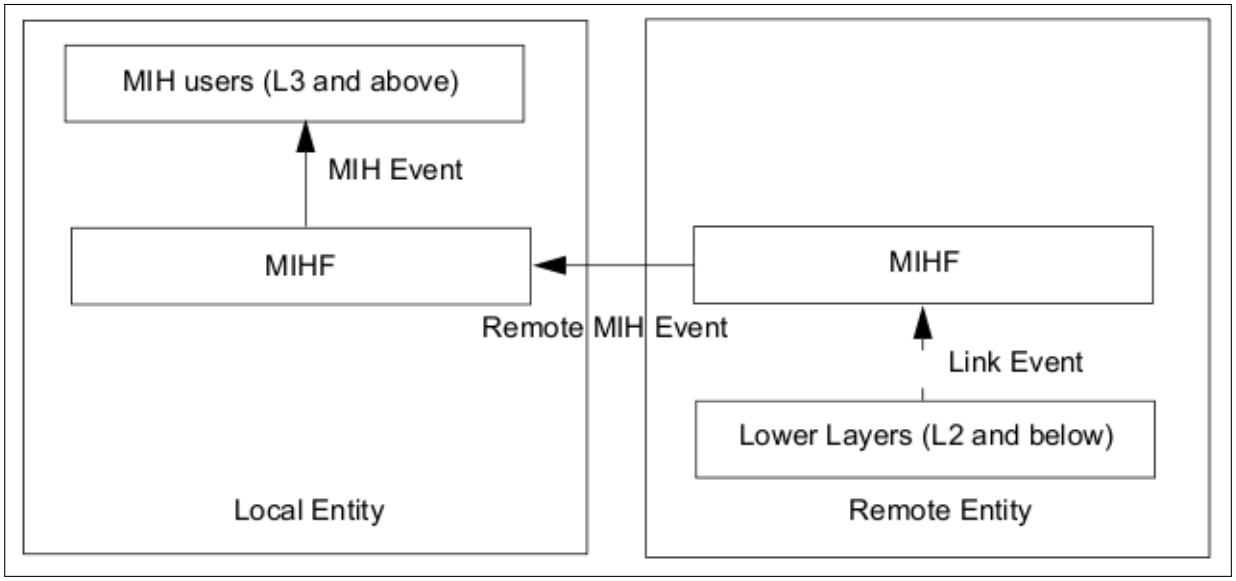

Figure 3.7: The remote demo shows the ability of a MIH_USR to obtain event notifications that happen on a LINK_SAP located on another machine. From [3].

3.3 Field Testing Topology

The proposed field testing scenario, as depicted in Figure 3.8, was designed to incorporate the DVBRCS testing hub, located at DISA's testing and lab facilities in Ft. Meade, MD, and two test suites located at an antenna farm, also at Ft. Meade, MD. In this scenario, an MIHF, including the MIIS and MIIS database, were to be located on a laptop (MS) at the hub, while two additional laptops (M1 and M2) that were to be located at a suite at the antenna farm. The Link_SAP on M1 would provide the connectivity to which the MIH user and MIHF on M2 would connect via an ad-hoc wireless connection. M2 would then be able to communicate with the MIHF on the MS at the hub.

The hardware features of each laptop used, are as follows:

Laptop at hub (MS)

- Type: Dell XPS 1530
- Operating System: Ubuntu 11.10
- RAM: 4 GB

Laptop at suite (M1)

- Type: Lenovo T400
- Operating System: Ubuntu 11.10
- RAM: 2 GB

Laptop at suite (M2)

- Type: Lenovo T61
- Operating System: Ubuntu 11.10
- RAM: 2 GB

In this scenario, we also used the Kamailio SIP server and Twinkle SIP client as our mechanism for conducting handover, by starting and maintaining a connection between the SIP server and a SIP client, while making attempts to move from network to another. In order to do this, the SIP connection should be started prior to ODTONE. Other options for this scenario included the UMIP Mobile IPv6, a proxy Mobile IPv6 implementation, and Host Identity Protocol (HIP), which we recently learned is an implementation that can be successfully used in conjunction with ODTONE for handover.

3.4 Field Testing Topology 2

After testing the scenario in the first field test, we determined that, while this test proved some validity, it was not robust enough to suit our needs. We decided that we required a topology that was more realistic in terms of what would have to be set up to test ODTONE within the framework of DVBRCS. We then designed the following network that would place each node in its own subnet, vice the single subnet network that we designed for the previous testing.

The configuration consisted of two laptops, both configured with ODTONE; a CISCO 2600 router; an ASUS wireless router (model RT-N56U); and a Linksys wireless router (model WRT54GL). In this scenario, we used different laptops that we preconfigured for this testing scenario, rather than the laptops that we used in the previous test. The hardware features for

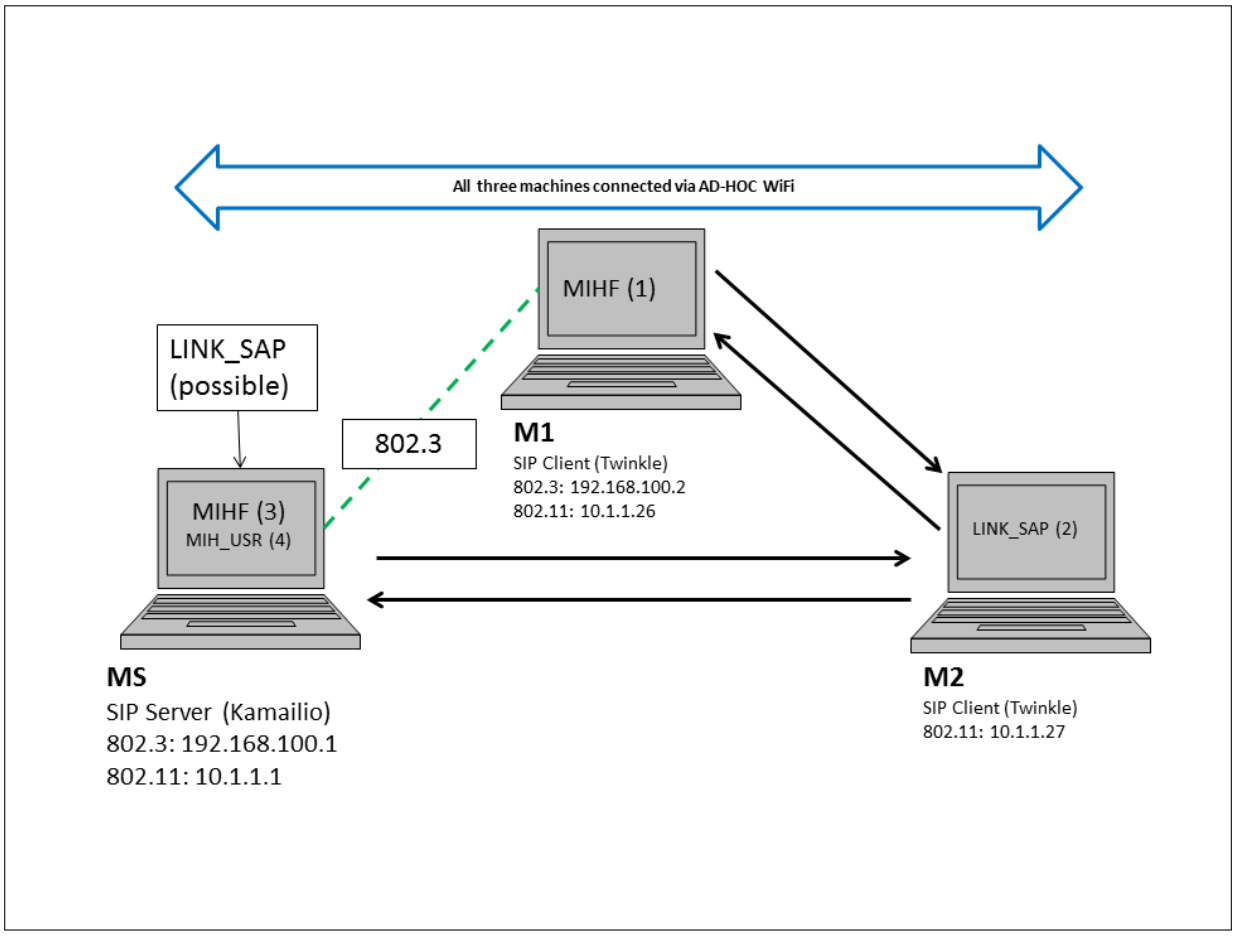

Figure 3.8: Testing scenario used in our initial testing environment, in the absence of connectivity to the DVBRCS suite.

each of the laptops were as follows:

Laptop at suite (M1)

- Type: Dell XPS 1530
- Operating System: Ubuntu 12.04 LTS
- RAM: 4 GB

Laptop at suite (M2)

- Type: Dell Latitude D830
- Operating System: Ubuntu 12.04 LTS
- RAM: 2 GB

We configured one laptop (M2) with two wireless NICs and a SIP client to maintain connectivity to the Kamailio SIP server, which was located on the other laptop (M1) and was connected to the CISCO router. M1 also was configured with an MIHF and the ODTONE SAP_8023 LINK_SAP. During this test we connected M1 to a Fast Ethernet port on the CISCO router, sttarted the Kamailio SIP server, MIHF1, and the SAP_8023 LINK_SAP. The ASUS and Linksys wireless routers were configured and connected to Ethernet ports coming off of the CISCO router. MIHF2 on M2 was then started along with the two 802.11 LINK_SAPs (one for each wireless NIC), once these services were started, we established our SIP connection between the Twinkle SIP client on M2 to the SIP server on M1, and then started the MIH_USR.

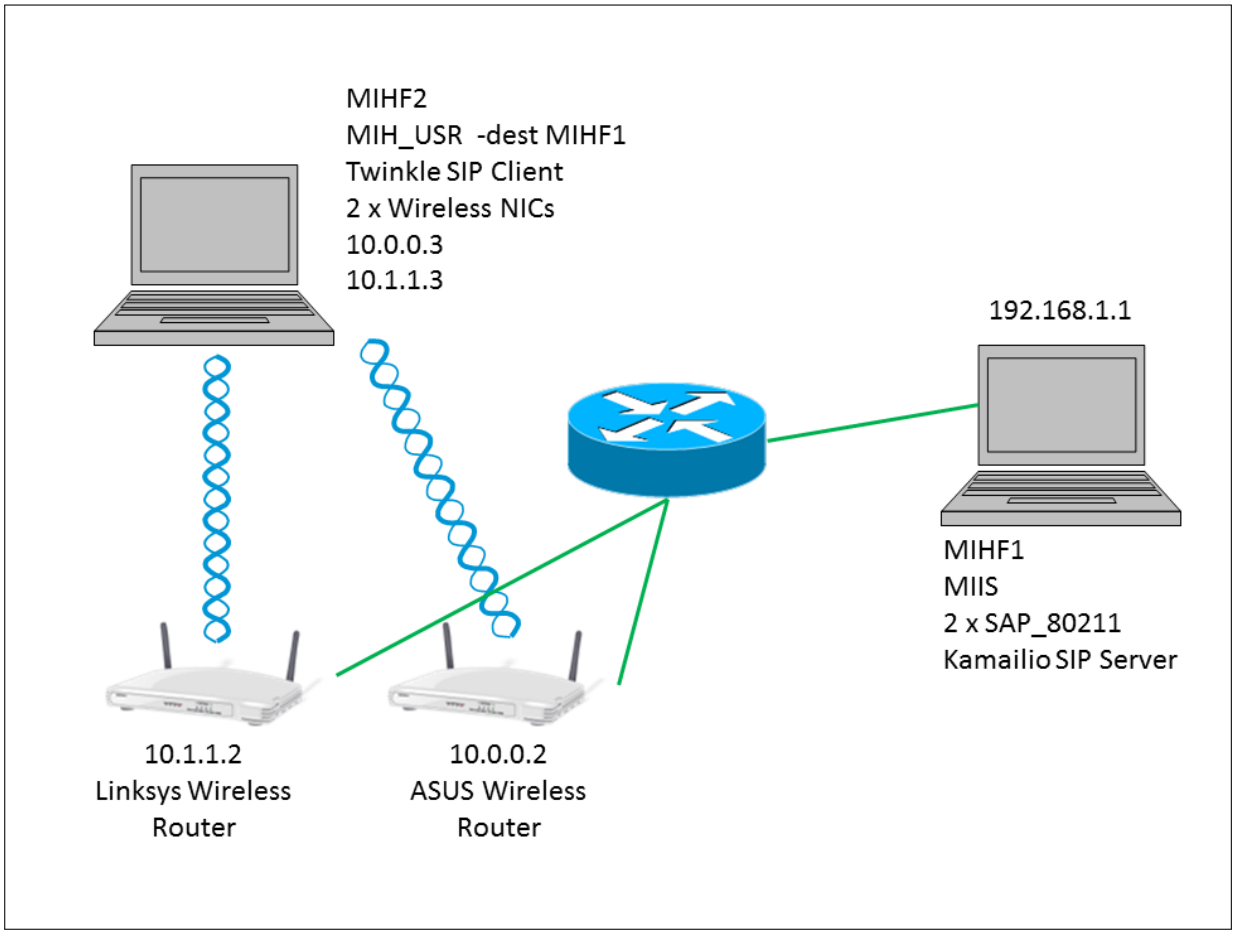

Figure 3.9: Architecture used in the final testing scenario, designed to test ODTONE's SAP_80211 LINK_SAP.

These scenarios, with the SIP study included, allowed us to gain a better understanding as we attempted to implement this open-source solution of MIH. ODTONE's capabilities provided us

with the mechanisms to conduct handovers between networks by presenting a valid MIHF for us to test. The results of these testing scenarios are presented in the next chapter.

CHAPTER 4:
Results

This chapter examines and provides an analysis of the results that we gathered and generated during all phases of this research. It specifically examines the results gathered and determines the relevance of those results to the DISA 802.21 effort as well as their significance to future implementations of Media Independent Handover services within the framework of DoD mobility and full-motion video dissemination systems.

4.1 ODTONE Demonstration Results

The execution of the scenario outlined for the demonstration provided the results we expected. In both the local and remote demos, we saw several event notifications consistent with the documentation. This is a significant finding, as it confirms the implementation of the demo LINK_SAP in ODTONE's local and remote demonstrations. Using Wireshark to isolate the MIH packets, we noticed several instances of both the 802.11 link and the 802.3 link showing LINK_UP and LINK_DOWN when we toggled the wireless switches on both laptops off and on in the remote demonstration. This initial testing caused some alarm, as we were expecting to receive either LINK_UP or LINK_DOWN notifications when conducting handover from one network to another, and did not expect to see the links going up and down at the same time. After consulting with the developers, we found that this occurrence was due to the code not isolating each of the LINK_SAPs for these heterogeneous networks. This was something that must be addressed by manipulating the code in the LINK_SAPs and could not be accomplished by simply adjusting the configuration files for both of the LINK_SAPs. Unfortunately, this was something that we discovered too late in the study to address by this thesis, though it would certainly have resulted in a significantly deeper understanding of the code. However, this issue was in fact only a peculiarity of the demonstration version of the software and not of a production LINK_SAP, which we were able to get from the developers for the final testing conducted at DISA. The results of the latter testing are presented later in this chapter.

After reviewing related work, we expected to get similar results to those experiments conducted in a lab setting. Since our initial testing was also done in a lab, we did not expect our results to vary in any way. Once our testing environments changed and we were using real-world systems, we expected the results to differ slightly, depending on the protocol used for handover. Of the

three protocols supporting handover, we were not able to find results from the literature about handovers conducted using the Host Identity Protocol. This could be because this protocol is still in an experimental phase and has not been fully vetted. The other two protocols, SIP and Mobile IP, have been tested and have provided different results, with regards to handover time.

One of these studies focused only on the use of SIP with a WLAN to WiMAX handover. This experiment was conducted as a joint collaboration between British Telecom and Intel. They were able to get the handover time down to approximately 305 ms with no packet loss, as the new connection (WiMAX) was made before the old connection (WiFi) was broken. [13] These results established a baseline for our exploration of SIP-initiated handovers, which hinged on ODTONE working properly.

4.2 Field Testing Results

In retrospect, we determined that our experiment setup would not address some of our essential information elements. In particular, we were limited by our ability to use only the SIP protocol for handover management, as well as limitations on the sourcing of the video stream. Due to DISA security policies, we could not connect the MS laptop to the DVBRCS hub itself, instead the MS laptop had to be connected to the second suite that was available at the antenna farm. Exception to the policy DISA had in place for the hub must be approved well in advance because the hub is part of the Tier III help-desk for the live system that is currently on-line in the CENTCOM AOR. Another issue encountered was if we were to initiate an event trigger that would cause the MIH user on laptop M2 to handover from an Ethernet connection (802.3) to a WiFi connection (802.11) we would lose the video feed that was being streamed across the suites, because we had no means to distribute video without a switch or router at the antenna farm. This was an oversight, in that we did not arrange to have this equipment available for our testing. The only way that we could have made an immediate impact to this situation was by adding a wireless router to the interface on the suite to which we were connecting at the antenna farm. However, due to local security policy, such a configuration change could not be made on the fly, and thus would not be available within the time constraints of the experiment.

While MIH allows for any of three different protocols to execute handover, in our testing scenario, as seen in Figure 3.8, we were only able to use SIP as a means for conducting handover, due to equipment and code limitations of HIP and Mobile IP (MIP) sources; particularly a valid MIP functionality. We did not have an external source that could provide MIP to our system. During the installation phase of ODTONE, and in the initial testing, we were unable to find a

stable version of either protocol. While we were told that HIP was an easy protocol to use for the testing, due to its ability to secure a permanent IPv6 address bound to the host machine, the stability of the available software was limited. Mobile IP and Mobile IPv6, would have been optimal for testing due to their ability to negotiate handover quickly; however, our testing laptops had problems with the IP mobility functionality in the lab environment. The lack of HIP and MIP source was not altogether problematic, but the non-availability proved to be a constraint for which we had not planned. Due to limitations in the availability of stable HIP and Mobile IP source code, we were not able to complete a successful test of MIH using anything but SIP.

We also made the decision to use only SIP to conduct handovers between MNs. While this is not considered the optimal protocol to use for conducting handovers, it was the only protocol we were able to get working in the configuration used for our experiments. The issue here is a handover conducted using SIP takes about 350 ms, while a handover using MIP will take around 150 ms. According to a brief given by Applied Communications Sciences, the optimal handover time should be 250 ms or less, which places SIP at a performance disadvantage. [14]

We relied on two open source platforms for SIP functionality, the Kamailio SIP server and the Twinkle Softphone/SIP client, both supporting Linux. In all testing instances, after the MIH connection was validated, we initiated a SIP session from M2 to M1 via the Twinkle softphones. With a validated connection, we were able to pass voice and instant messages back and forth between the two clients, allowing for Real Time Protocol (RTP) sessions to be maintained.

The ODTONE software presented some challenges that can only be attributed to limitations in the available documentation and to the fact that ODTONE is compatible with Windows as long as the Windows kernel is VISTA (NT 6.1) or later. Since our laptops were running the Windows kernel version NT 5.1, we were required to set up all of our clients and the server in a Linux environment. We used Ubuntu 11.04 and 11.10, but further evaluation showed that ODTONE could also be used in Ubuntu 12.04 LTS. While it took several attempts to get ODTONE working, we were eventually successful. The issues that we faced were primarily linked to the limited documentation provided by ODTONE, work-arounds related to the ODTONE source code, and to problems that we faced with the general understanding of the Boost software that is used to compile the ODTONE source. However, once the experiment resources were functional, we were able to gather sufficient data to evaluate the capabilities provided by the ODTONE software.

We decided that the best way to evaluate the data would be by using Wireshark, with the mod-

ified dissector that is available through the Wireshark bugs database website [15]. Wireshark simplified the capture and evaluation of any MIH packets generated by communication between the LINK_SAP, the MIH_USR, and the MIHF, including the MIES event notifications. Some of the initial results were not populated in Wireshark se we had to rely solely on log files that we pulled from the LINK_SAP and MIHF terminal windows on the laptop to collect such data. These files enabled us to match up events that we saw annotated on the terminals, while the components were running, with the packets that we saw in Wireshark.

In order to trigger these events, all of the test machines (MS, M1, and M2) were wired together through an Ethernet (802.3) hub, with MIHF1 on laptop MS, MIHF2 on laptop M1, and the LINK_SAP on laptop M2, and were also connected wirelessly via an ad-hoc wireless network. Our initial test was to cause a handover from an 802.3-connection to an 802.11-connection and then back to the 802.3-connection. Using SIP as our handover protocol, as noted above, we established an instant messaging connection between the two MIH_User connections. The flow of instant messages between the two machines established a Real-Time Protocol (RTP) session between them. This allowed us to "see" if our connectivity was maintained by looking at the RTP packet flow within the SIP connection via the Wireshark packet captures.

In general, we had to make several attempts to stimulate the generation of MIH packets by manually triggering events such that the packets could be observed with Wireshark. Such stimulation included manually toggling the 802.11 connection at the laptop (M1) running the LINK_SAP configured for the wireless NIC or by disconnecting the 802.3 connection on either laptop (M1 or M2). We also launched a SIP session with the Kamailio SIP on laptop MS server and the Twinkle SIP client on laptops M1 and M2 throughout the testing.

This testing generated more than 150,000 packets through the course of the experiment. Using Wireshark and a dissector, turned into a patch, we were able to isolate events created by the MIES as it communicated through the MIHF. This provided us the ability to see when the MIHFs were communicating with each other to establish a connection, and how they responded to the link event (up or down). In Figure 4.1, the first six packets show this in action:

1. (from MS/MIHF1 to M1/MIHF2) Service Management Request "MIH Capability Detected"
2. (from M1/MIHF2 to MS/MIHF1) Service Management Response "MIH Capability Detected"
3. (from MS/MIHF1 to M1/MIHF2) Service Management Request "MIH Event_Subscribe"

4. (from M1/MIHF2 to MS/MIHF1) Service Management Request "MIH Event_Subscribe"
5. (from M1/MIHF2 to MS/MIHF1) Service Management Response "MIH Event_Subscribe"
6. (from MS/MIHF1 to M1/MIHF2) Service Management Response "MIH Event_Subscribe"

Since we were using SIP as our means of handover, SIP packets can also be observed in the packet capture showing when SIP connections were made and, in some cases, broken but later repaired.

1. (Packet number 9) Request: REGISTER SIP:192.168.100.1
2. (Packet number 10) Status: 400 Forbidden (0 bindings)
3. (Packet number 17) Request: REGISTER SIP:192.168.100.2
4. (Packet number 18) Status: 200 OK (1 bindings)

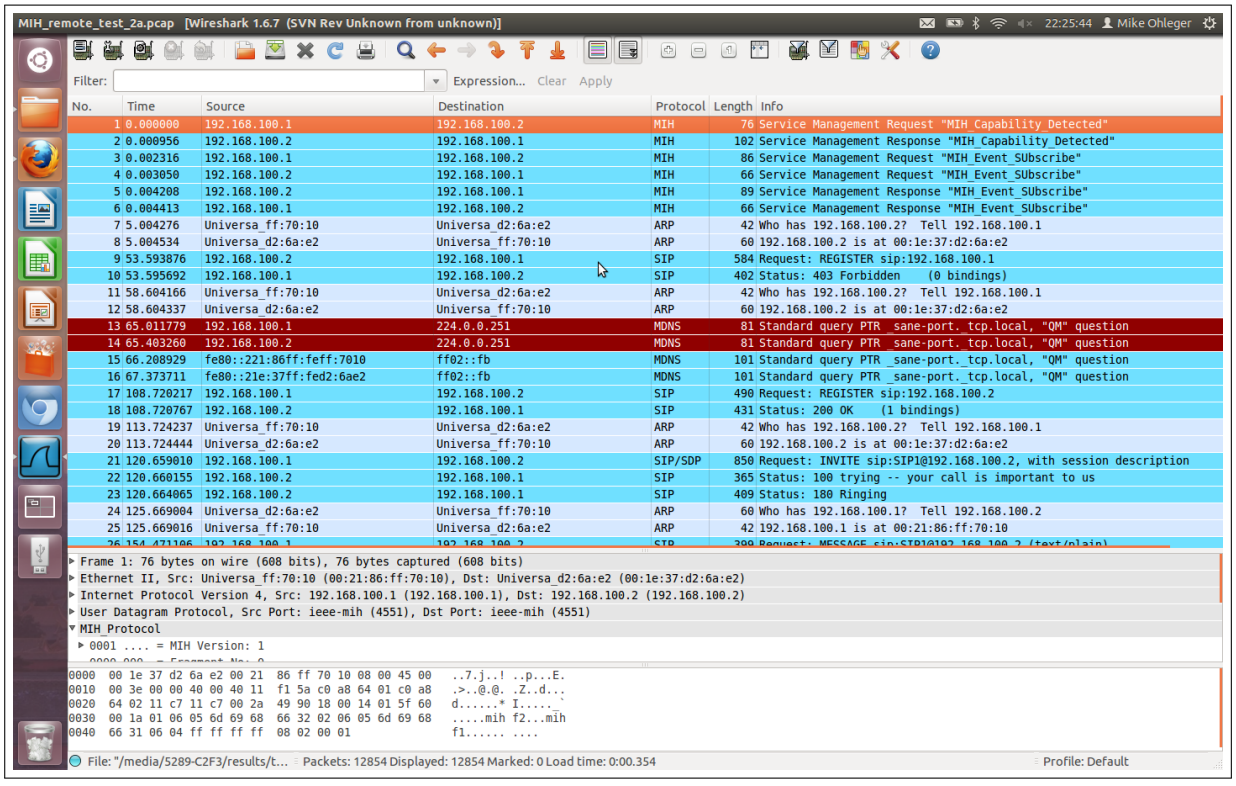

Figure 4.1: Screenshot of Wireshark packet capture, isolating MIH and SIP packets.

In our analysis of the traces we collected via Wireshark (Fig. 4.1), we were able to see triggered events through the MIES, such as MIH_Capability_Detected, MIH_Event_Subscribe, MIH_Link_Down, and MIH_Link_Up. Each of these events was significant in that they were generated when each MIHF communicated with another via the MIH_Link_SAP and the MIH_User.

We then attempted to trigger handover events. When the wireless radios were turned off and on, the connection was maintained via the 802.3 links; however, the notifications we saw indicated that there was an MIH_Link going up or down. We could only infer directly that despite the loss of the 802.11 connection we were, in fact, switching between the two heterogeneous 802 networks with little to no loss of connectivity. We saw no degredation or loss of signals and observed little to no packet loss in the SIP session between the Kamailio SIP server and the Twinkle SIP client during the handover process. However, we came to realize that this particular test was not as successful as we had originally thought, because we could not necessarily prove that a seamless handover occurred, and because the SIP connection between the server and client was unreliable on an ad-hoc network. Further testing with the distribution LINK_SAPs in ODTONE also allowed us to discover that the code written for the LINK_SAPs only provided event notifications to the MIHF and was not designed to be used for anything but demonstration purposes. These LINK_SAPs were designed to only trigger events within the MIHF. Regardless of the fact that we had done major configuration changes to the MIHF, the LINK_SAP, and the MIH_USR on each individual machine, the code was not intended to actually conduct handovers.

4.3 Field Testing 2 Results

In the final testing scenario, we tested two additional LINK_SAPs of ODTONE of which we were not initially aware: the SAP_80211, and the experimental SAP_8023. Each of these LINK_SAPs was designed to be an actual working PoA for an MIHF. For the final testing scenario, we were able to test them and found some interesting results. The ODTONE configuration files used for this testing scenario can be found in Appendix A. The drawback of this testing, however, was that we were not sure how to configure the SAP_8023 correctly, and only used it as the PoA for the host machine (MIHF1) that was supporting the SIP server. The overall design and functionality of the SAP_80211 was created to interact in the same manner as the demo LINK_SAP with the MIHF and the MIH_USR, thus generating trigger events similar to those found in the demos.

For this scenario we set up two wireless routers, one inside the building (ASUS Router) and the other (Linksys Router) outside the building using a 30 foot LAN cable so we could maintain connectivity to the CISCO 2600 Router. We then took the laptop with the two wireless NICs outside, and walked around the area in order to move from one wireless network to the other. We felt that this would be the most realistic scenario to support handover from one network

to the other. Attachments to individual networks were monitored based on signal strength and connectivity to the SIP server. In some instances, while we were able to see the same signal strength being broadcast from the individual wireless APs, the MIHF continued to choose the network based on signal strength. At the same time, we were able to see other networks that were available, mostly due to the fact that there were barracks nearby that had wireless APs broadcasting their SSIDs. While we were in range of these networks, we assumed that we would possibly have been able to connect to them if they were open networks and if we were enabling DHCP within ODTONE.

What we discovered, however, was that we could not see all of the MIH packets through Wireshark. Therefore, we were required to view and manually analyze the data generated by the individual LINK_SAPs, primarily with the ODTONE-developed sap_80211 LINK_SAP. When running these particular LINK_SAPs, we were able to save the log files that were generated in the open terminal session. During this testing session, we watched the LINK_SAP negotiate several different wireless networks before it attached to an existing PoA, thus establishing a connection for handover. The only limitation that needs to be considered for this final scenario is that we limited our testing to static networks running on two separate wireless NICs, since we were not using the MIIS for our testing, and could therefore not implement ODTONE's DHCP component. We did, however, take note of the fact that as we moved out of range of one of the wireless routers, the LINK_SAPs negotiation of the two available networks happened quickly enough that we were able to attach to the new network while still maintaining the SIP connection between the Kamailio server and Twinkle client. Each of the LINK_SAPs conducted a search of available networks, found valid networks, and attached to the network that had the stronger signal, as we had expected they should. Figure 4.2 and 4.3 contain the log files captured from the two separate SAP_80211 LINK_SAPs we were running on the client machine (M2). Each log file shows the negotiation with available networks and the subsequent attachments to valid PoAs. As mentioned previously, these PoAs were running static IP addresses, so the networks discarded in the process were addtional PoAs that were candidates but were not in the MIIS database, and therefore, no attachments could be made.

Of particular interest in the log files taken from the 10.0.0.2 network running on the ASUS router is the indication the connection is dropped once we moved out of the range of the router. The SAP_80211 for that network made attempts to negotiate additional networks, but was not able to find a valid PoA to which it could attach. At the same time, the SAP_80211 on the Linksys network (10.1.1.2) negotiated and maintained a valid connection throughout this sequence of

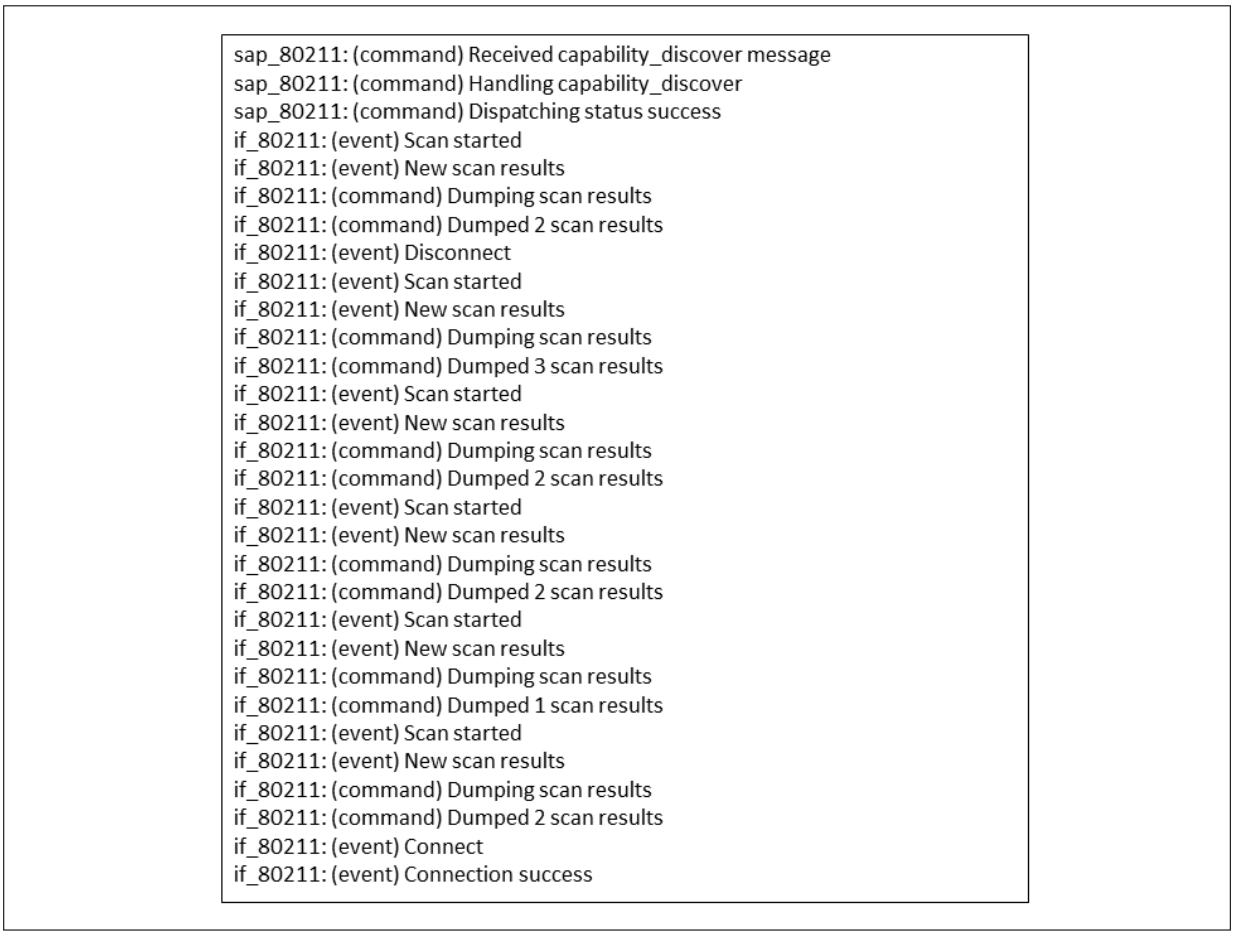

Figure 4.2: Log file taken from 10.0.0.2 network on M2, that was running on an ASUS WiFi router. This log file was generated by pushing the information that was appearing in the terminal window to a .txt file (./sap_80211 » asuslog.txt).

events. Once we moved back into the range of the ASUS router, we were able to reestablish connectivity to the network.

In this final test, we made several observations that we were not able to make previously. Most particularly, we had evidence that we had, in fact, conducted a handover between two different 802.11 networks without dropping any packets and were able to maintain a SIP connection between the SIP server on MIHF1 and the SIP client located on MIHF2. Through this SIP connection we ensured that we maintained a constant flow of RTP packets between the client and the server, via instant messaging. While this test appeared to be fairly accurate, we did notice that some instant messaging attempts resulted in failures. This could have been caused by our rate of movement as we negotiated the networks; however, we are not positive that this

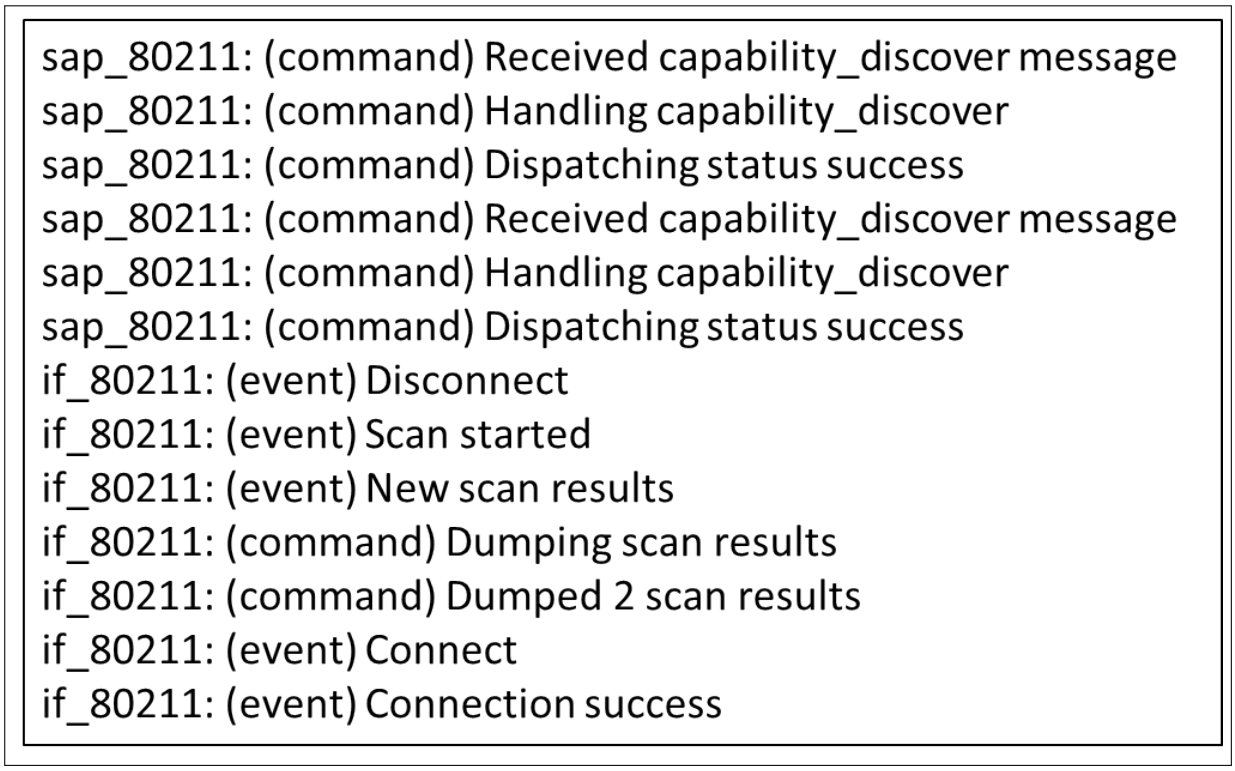

Figure 4.3: Log file taken from 10.1.1.2 network on M2, that was running on a Linksys WiFi router. This log file was generated by pushing the information that was appearing in the terminal window to a .txt file (./sap2_80211 » linksyslog.txt).

was the case. Nonetheless, we did not observe any packet loss when analyzing the network traces we collected in Wireshark. Figure 4.4 contains a screen capture showing what we believe to be a successful handover between the 10.0.0.2 network and the 10.1.1.2 network (the 192.168.1.2 destination IP address is that of the instant messaging hosts), with a handover time of approximately 500 miliseconds, which is the expected handover rate for handovers using SIP [14].

Due to these findings, we can make reasonable assumptions that if we had been able to test streaming video during this final testing, we would have been able to successfully handover between the two networks without dropping any packets in the video stream.

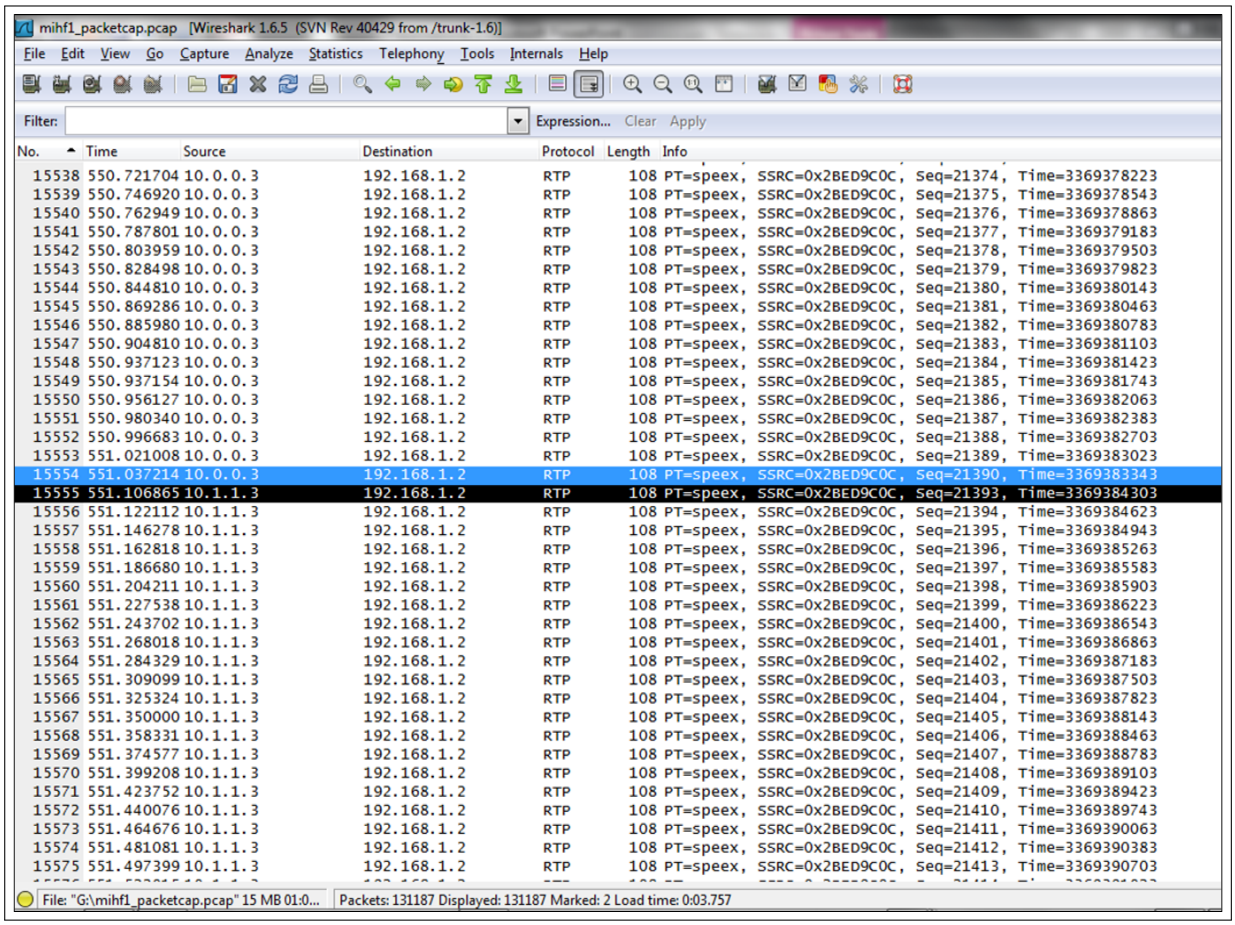

Figure 4.4: The first handover between the 10.0.0.2 network and the 10.1.1.2 network occurred at packet number 15555 (out of approximately 150,000 total packets).

CHAPTER 5:
Conclusions and Future Work

This chapter discusses conclusions drawn through research and experimentation of open source 802.21 implementations, in this case Open Dot Twenty ONE (ODTONE). It also discusses work that should be accomplished in the future in order to implement the technology within the existing DoD communications framework, as well as the benefits that this technology will bring to deployed units.

5.1 Conclusions

Through testing and research of the 802.21 standard, we have found that it is a viable technology that could provide DISA with the means to leverage wireless networks to extend UAV video dissemination beyond their current capabilities. The amount of preliminary testing we conducted shows the potential for growth in this area thereby validating its validity as a technology for DISA to pursue further. While we were not able to answer all of the research questions we identified at the outset of this effort, it is clear that our study was limited by the current state of implementations and not by the standard itself. While MIH is yet to be widely adopted by the commercial sector, in terms of mobile service providers like Verizon and AT&T, the applications MIH would enable within DoD communications is plentiful. Should DISA, and DoD by extension, adopt MIH as an operational standard, DISA should promote the development of additional and more advanced MIH implementations.

While we had some success employing and testing ODTONE, there were some components of the software that we were not able to fully test. The ODTONE suite is able to support a solid test bed for future work; however, a full working model of this software has yet to be developed and tested. The limitations that we discovered while working with ODTONE provided several challenges that we were not able to overcome in the preliminary testing environment we implemented. We were not able to test the advertised capabilities of 802.21 with ODTONE due to our own lack of developmental capabilities. As an open source project, ODTONE is designed to allow for development of new LINK_SAPs such as 802.16, 3G, LTE, etc. by the developer community at-large. While we were able to procure an experimental version of the 802.3 LINK_SAP that was developed by ATNOG, we are not certain that our testing provided any proof that the provided LINK_SAP worked in the way that it was expected to perform.

We were also not able to test the MIIS functionality properly as we did not have the expertise to build and develop a database that would support the framework as outlined by the standard. Unfortunately, the ODTONE documentation was not clear as to how to build the database needed to support this important piece of the MIHF framework. The ODTONE documentation does mention that while the developers had supplied an experimental database, the user could provide whatever one they would like to use.

Ultimately, if DISA were to adopt and pursue the IEEE 802.21 standard as a primary means for extending networks and pushing them beyond the limits of the FOBs, the result could be of great benefit to the troops on the ground. Troops on the ground would have access to information at a moment's notice and in near real-time, thus improving the reach, reliability, and applicability of DVBRCS and the DDAN. One alternative to MIH is to continue to conduct operations as usual. This alternative would lower the situational awareness and could be a hinderance to the soldiers and Marines that require real-time information which could lead to loss of life. An additional alternative is to use the 802.11u technology. 802.11u is a newly published amendment to the IEEE 802.11 standard that allows for mobile handoff between 3G and WiFi networks that are not pre-authorized and that allow access based on the user's relationship with an external network [16]. While the IEEE 802.11u standard will work as a stand-alone technology, it is limited to handover between 3G and WiFi networks only. This standard provides a common abstraction that would allow devices to use 802.21 to handoff between 3G and WiFi networks, by providing a means for common authentication, regardless of protocol used [16]. The best possible solution for scenarios involving WiFi would be a combination of 802.11u and 802.21.

A recent update to the IEEE 802.21 standards document (DRAFT 802.21b) [4] outlines the changes to the standard to include information on extensions for supporting handovers with downlink only technologies. This update introduces a new command service flow that includes the downlink only technologies, as seen in Figure 5.1.

In light of this new information, the standard is designed to support streaming media and, while we were unable to test this due to time constraints, it is important to note that the IEEE is making efforts to support this functionality. With the plethora of streaming media technologies coming on line, 802.21b is certainly destined to have an immediate impact on DISA's continuing efforts with DVBRCS and the DDAN.

5.2 Future Work

Additional testing designed to test cellular network (3G, 4G, LTE), WiMAX, and CDMA/GSM handovers should continue over the near foreseeable future to further prove the technology.. The recommended testbed, which we designed and as depicted in Figure 5.2, should be setup in a lab environment to provide a more comprehensive experiment allowing for the introduction of Mobile IP as a handover method, Femtocells to be used to test 3G handover, a WiMAX Rapidly Deployable Network (RDN), or other suitable WiMAX device, that will allow for the testing of 802.11 to 802.16 handover, and an Android smartphone to use within the 802.11-to-3G handover. Upon successful completion of this portion of the test the previous architecture would need to be incorporated back into the DVBRCS framework in order to test the capabilities of the DVBRCS hub and suite along with broadcast video. This testbed environment will allow the testing of all available functionality for MIH using ODTONE. A key limitation to this testing scenario is that ODTONE currently will only support an 802.11 LINK_SAP and an experimental 802.3 LINK_SAP. With development within the open source community on-going, we expect additional LINK_SAPs.

While this testbed network is based on the use of ODTONE, other software implementations of the standard should be considered, especially as additional handover needs emerge in the future.

Additionally, if 802.21 is to be implemented within DOD, there will need to be further testing and evaluation in two specific areas of importance that were not covered in this study. Those fields of study are MIIS investigation and functionality, as well as the security aspect of the standard.

5.2.1 MIIS

The MIIS is an important component of MIH, as it is designed to reduce handover latency by discovering information beforehand about the available networks within a geographical area and providing that information to the mobile nodes on demand. Without this component, the MIHF may not completely function as intended since it will not be able to discover alternate networks to serve as a PoA. While we made attempts to implement the MIIS in our testing and evaluation, we were not able to perform a successful test of MIIS integration with the MIHF in its entirety due to our lack of understanding of how it was implemented within ODTONE. While knowing that the MIIS is only required if network discovery is necessary, and will not be used if candidate networks are identified by static IP address schemes, a demonstration of its current capability in a controlled test environment should be performed. Additionally, a

suitably comprehensive evaluation of the current state of the art MIIS should be designed and implemented.

5.2.2 Security

The security aspect of 802.21 appears to be a hard problem that will require more time and effort to evaluate and understand. With the various security implementations available in wireless networks, there are several challenges that arise even when trying to conduct handovers between the quasi-homogeneous 802.11 networks. Items for consideration include, but are not limited to,TKIP and WPA2. Each of these wireless security types has its own requirements for authentication that will require input into the MIIS database. If the MIIS is to be a self-populating database, there exists a problem with the passphrases and passwords that will have to be manually entered into the database for access control. As the database begins to grow with respect to the number of entries, so does the requirement for providing an automated means for authentication. Further, the database itself will require encryption as well. As we consider additional heterogeneous networks (802.3, 802.16, 802.22, etc.) and cellular networks (3G, 4G, LTE, EDGE) and even CDMA and GSM, we must also consider the encryption means supported by each technology. Therefore, in order to completely comprehend the security aspects, it is also important to understand how the MIIS works and how it is best implemented. This is further impetus for exploring the MIIS concept in depth.

These are but a few of the issues remaining to be addressed or considered as DISA continues its consideration of the Media Independent Handover construct. The technology offers great advantage and bears exploration. This thesis serves as one first step in that exploration.

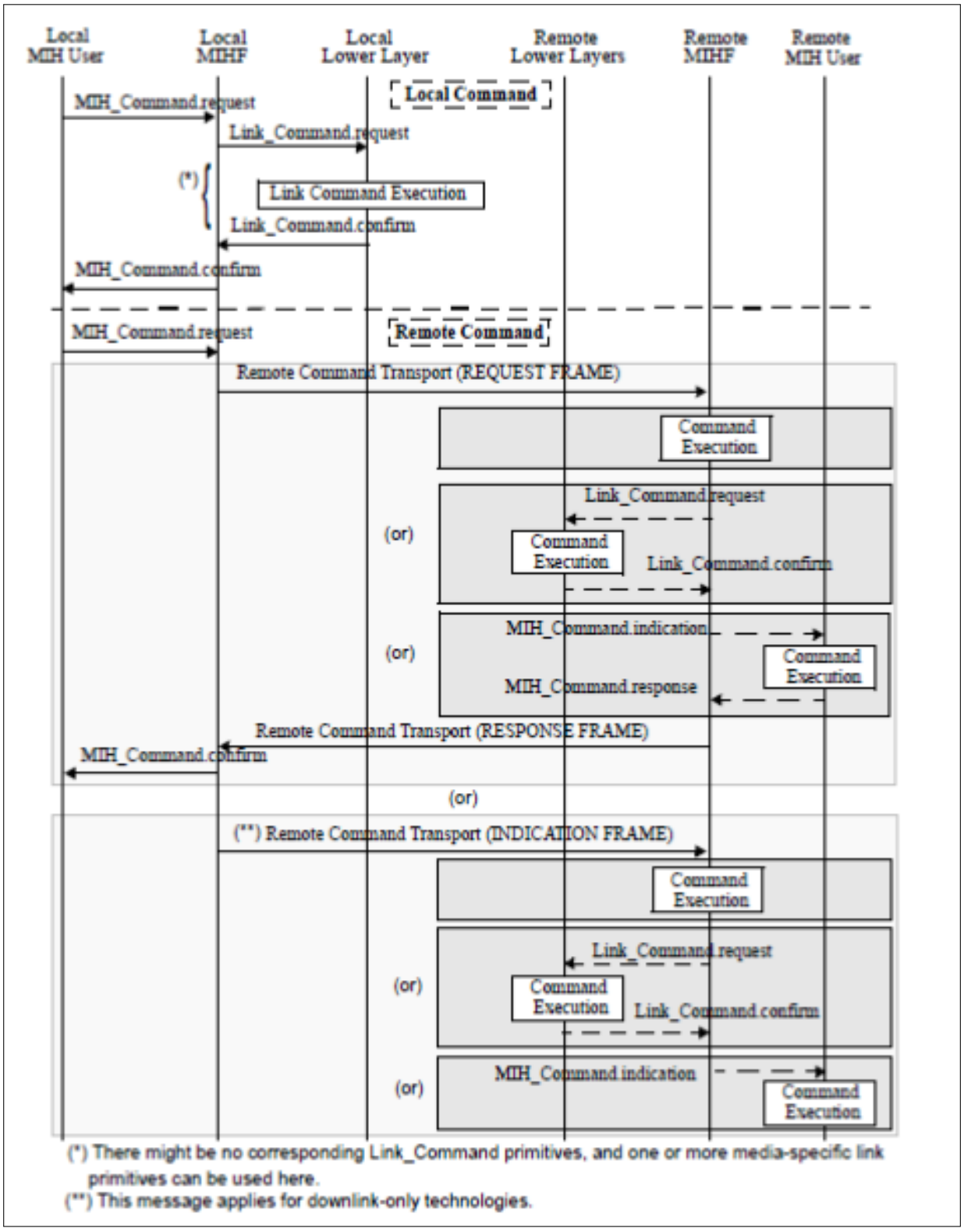

Figure 5.1: IEEE 802.21 Command Service Flow with additional Remote Command Transport for downlink only technologies. From [4].

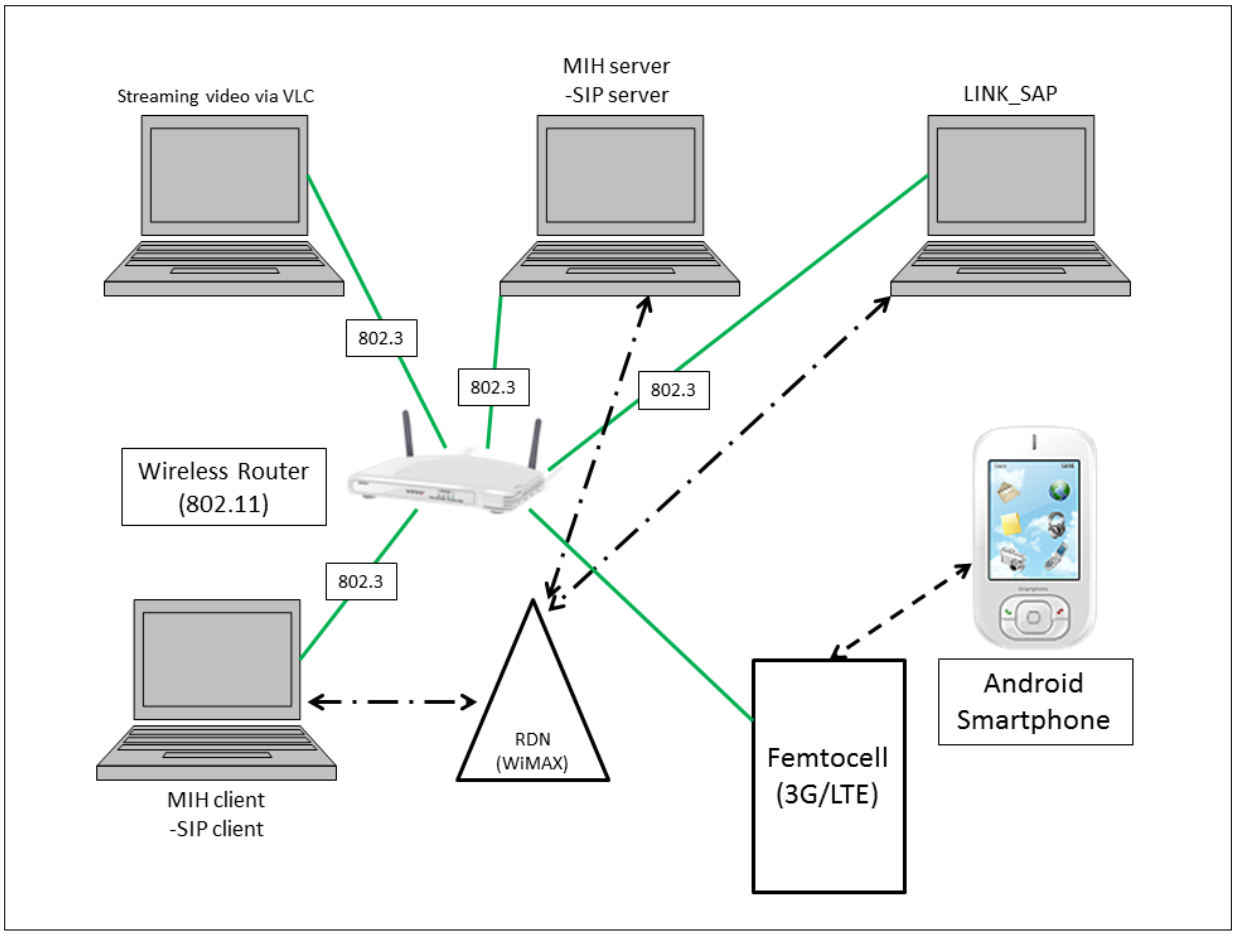

Figure 5.2: Planned testing scenario for in-house testing of 802.21, with architecture similar to what should be considered as a part of the final testbed with the DVBRCS hub and suites.

Appendix: ODTONE Configuration Files

This Appendix provides all of the configuration files we used in our field test 2 as outlined in Chapters 3 and 4. MIHF1 served as the host machine that used the 802.3 LINK_SAP, and MIHF2 was the client machine that was equipped with two wireless NICs and the two 802.11 LINK_SAPs.

MIHF1

ODTONE.conf

```
#============================================================================
# Brief   : MIHF configuration file
# Authors : Carlos Guimaraes <cguimaraes@av.it.pt>
#----------------------------------------------------------------------------
# ODTONE - Open Dot Twenty One
#
# Copyright (C) 2009-2012 Universidade Aveiro
# Copyright (C) 2009-2012 Instituto de Telecomunicações - Pólo Aveiro
#
# This software is distributed under a license. The full license
# agreement can be found in the file LICENSE in this distribution.
# This software may not be copied, modified, sold or distributed
# other than expressed in the named license agreement.
#
# This software is distributed without any warranty.
#============================================================================

[mihf]
##
## This mihf's id
##
## Usage: id = <MIHF ID>
##
```

```
id = mihf1

##
## Port on localhost that MIH Users and MIH Link SAPs connect to.
##
## Usage: local_port = <port>
##
local_port = 1025

##
## Port to which remote peer MIHF connect to
##
## Usage: remote_port = <port>
##
remote_port = 4551

##
## Comma seperated list of remote MIHF's
##
## If you want to test remote MIHF communication add an entry here
## with the IP address of the remote MIHF.
##
## Usage: peers = <mihf id> <ip> <port> <transport protocol list>, ...
##
peers = mihf2 10.0.0.3 4551 udp, mihf2 10.1.1.3 4551 udp

##
## Comma separated list of local MIH User SAPs id's and ports
##
## Usage: users = <user sap id> <port>
[<supported commands> <supported queries>], ...
##
users = user1 7777
```

```
##
## Comma separated list of local MIH Link SAPs id's and ports.
##
## Usage: links = <link sap id> <port> <technonoly type> <interface>, ...
##
links = wired 11115 802.3 00:1d:09:39:fc:88,
wireless 51111 802_11 00:1d:e0:72:dd:79

##
## Comma separated list of the MIHF's transport protocol
##
transport = udp
```

MIH_USR.conf

```
#============================================================================
# Brief   : MIH-User configuration file
# Authors : Carlos Guimaraes <cguimaraes@av.it.pt>
#           Bruno Santos     <bsantos@av.it.pt>
#----------------------------------------------------------------------------
# ODTONE - Open Dot Twenty One
#
# Copyright (C) 2009-2012 Universidade Aveiro
# Copyright (C) 2009-2012 Instituto de Telecomunicações - Pólo Aveiro
#
# This software is distributed under a license. The full license
# agreement can be found in the file LICENSE in this distribution.
# This software may not be copied, modified, sold or distributed
# other than expressed in the named license agreement.
#
# This software is distributed without any warranty.
#============================================================================
```

```
##
## User id
##
[user]
id = user1

##
## Commands supported by the MIH-User
##
commands = mih_link_get_parameters, mih_link_configure_thresholds,
mih_link_actions, mih_net_ho_candidate_query, mih_net_ho_commit,
mih_n2n_ho_query_resources, mih_n2n_ho_commit, mih_n2n_ho_complete,
mih_mn_ho_candidate_query, mih_mn_ho_commit, mih_mn_ho_complete

##
## Port used for communication with MIHF
##
[conf]
port = 7777

##
## MIHF configuration. For the default demonstration leave as is.
##
[mihf]
local_port = 1025
```

SAP_80211.conf

```
[link]
##
## Link SAP identifier
##
id = wireless
```

```
##
## Link SAP listening
##
port = 51111

##
## Link SAP interface address
##
link_addr = 00:1d:e0:72:dd:79

##
## Scheduled scan period (milliseconds). Set to 0 to disable.
##
sched_scan_period = 0

##
## Default threshold checking period
##
default_th_period = 1000

[mihf]
ip=127.0.0.1
local_port=1025
```

SAP_8023.conf

```
[link]
##
## Link SAP identifier
##
id = wired
```

```
##
## Link SAP listening
##
port = 11115

##
## Link SAP interface address
##
link_addr = 00:1d:09:39:fc:88

[mihf]
ip=127.0.0.1
local_port=1025
```

MIHF2

ODTONE.conf

```
#============================================================================
# Brief   : MIHF configuration file
# Authors : Carlos Guimaraes <cguimaraes@av.it.pt>
#----------------------------------------------------------------------------
# ODTONE - Open Dot Twenty One
#
# Copyright (C) 2009-2012 Universidade Aveiro
# Copyright (C) 2009-2012 Instituto de Telecomunicações - Pólo Aveiro
#
# This software is distributed under a license. The full license
# agreement can be found in the file LICENSE in this distribution.
# This software may not be copied, modified, sold or distributed
# other than expressed in the named license agreement.
#
# This software is distributed without any warranty.
```

```
#============================================================================

[mihf]
##
## This mihf's id
##
## Usage: id = <MIHF ID>
##
id = mihf2

##
## Port on localhost that MIH Users and MIH Link SAPs connect to.
##
## Usage: local_port = <port>
##
local_port = 1025

##
## Port to which remote peer MIHF connect to
##
## Usage: remote_port = <port>
##
remote_port = 4551

##
## Comma seperated list of remote MIHF's
##
## If you want to test remote MIHF communication add an entry here
## with the IP address of the remote MIHF.
##
## Usage: peers = <mihf id> <ip> <port> <transport protocol list>, ...
##
peers = mihf1 192.168.1.2 4551 udp, mihf1 10.0.0.2 4551 udp,
mihf1 10.1.1.2 4551 udp
```

```
##
## Comma separated list of local MIH User SAPs id's and ports
##
## Usage: users = <user sap id> <port>
[<supported commands> <supported queries>], ...
##
users = user2 8888

##
## Comma separated list of local MIH Link SAPs id's and ports.
##
## Usage: links = <link sap id> <port> <techonoly type> <interface>, ...
##
links = wired 12345 802.3 00:1d:09:df:4f:d5,
wireless 54321 802_11 00:21:5c:32:0d:ad,
wireless2 22222 802_11 00:c0:ca:27:57:90

##
## Comma separated list of the MIHF's transport protocol
##
transport = udp
```

MIH_USR.conf

```
#============================================================================
# Brief   : MIH-User configuration file
# Authors : Carlos Guimaraes <cguimaraes@av.it.pt>
#           Bruno Santos     <bsantos@av.it.pt>
#----------------------------------------------------------------------------
# ODTONE - Open Dot Twenty One
#
# Copyright (C) 2009-2012 Universidade Aveiro
```

```
# Copyright (C) 2009-2012 Instituto de Telecomunicações - Pólo Aveiro
#
# This software is distributed under a license. The full license
# agreement can be found in the file LICENSE in this distribution.
# This software may not be copied, modified, sold or distributed
# other than expressed in the named license agreement.
#
# This software is distributed without any warranty.
#============================================================================

##
## User id
##
[user]
id = user2

##
## Commands supported by the MIH-User
##
commands = mih_link_get_parameters, mih_link_configure_thresholds,
mih_link_actions, mih_net_ho_candidate_query, mih_net_ho_commit,
mih_n2n_ho_query_resources, mih_n2n_ho_commit, mih_n2n_ho_complete,
mih_mn_ho_candidate_query, mih_mn_ho_commit, mih_mn_ho_complete

##
## Port used for communication with MIHF
##
[conf]
port = 8888

##
## MIHF configuration. For the default demonstration leave as is.
##
[mihf]
```

```
local_port = 1025
```

SAP_80211.conf

```
[link]
##
## Link SAP identifier
##
id = wlan0

##
## Link SAP listening
##
port = 54321

##
## Link SAP interface address
##
link_addr = 00:21:5c:42:0d:ad

##
## Scheduled scan period (milliseconds). Set to 0 to disable.
##
sched_scan_period = 0

##
## Default threshold checking period
##
default_th_period = 1000

[mihf]
ip=127.0.0.1
local_port=1025
```

SAP2_80211.conf

```
[link]
##
## Link SAP identifier
##
id = wireless2

##
## Link SAP listening
##
port = 22222

##
## Link SAP interface address
##
link_addr = 00:c0:ca:27:57:90

##
## Scheduled scan period (milliseconds). Set to 0 to disable.
##
## sched_scan_period = 0

##
## Default threshold checking period
##
## default_th_period = 1000

[mihf]
ip=127.0.0.1
local_port=1025
```

THIS PAGE INTENTIONALLY LEFT BLANK

REFERENCES

[1] IEEE, *IEEE Standard. 802.21 - 2008*. The Institute of Electrical and Electronic Engineers, Inc., 2009.

[2] B. Bennett and J. Walsh III, "Tactical services provider: media handover in heterogeneous communications systems," tech. rep., Defense Information Systems Agency, Ft. Meade, MD, 2010.

[3] ODTONE Software, available at http://atnog.av.it.pt/odtone/index.html.

[4] IEEE, *IEEE P802.21b/D06 Draft Standard for Local and metropolitan area networks*. The Institute of Electrical and Electronic Engineers, Inc., 2011.

[5] E. Piri and K. Pentikousis, "IEEE 802.21," *The Internet Protocol Journal*, vol. 12, pp. 7 – 27.

[6] J. Wong and V. Murahari, C.H. Cho, and V.C.M. Leung, "Secure media independent handover message transport in heterogeneous networks," *EURASIP Journal on Wireless Communications and Networking*, vol. 21, pp. 115 – 130.

[7] J. Stein, "Survey of IEEE 802.21 media independent handover services," 2006. Available at http://www.cs.wustl.edu/ jain/cse574-06/handover.htm.

[8] Claudio Cicconetti, Francesco Galeassi, and Raffaella Mambrini, "A software architecture for network-assisted handover in IEEE 802.21," *Journal of Communications*, vol. 6, pp. 44 – 55.

[9] W. Stallings, *Business Data Applications*. Upper Saddle River, New Jersey: Pearson Education, Inc., 2009.

[10] Series H: audiovisual and multimedia systems. Available at http://www.itu.int/rec/dologin_pub.asp?lang=e&id=T-REC-H.323-200912-I!!PDF-E&type=items.

[11] Kamailio - the Open Source SIP Server. Available at http://www.kamailio.org/w/.

[12] Twinkle - SIP Softphone for Linux. Available at http://www.twinklephone.com/.

[13] K. Choong, V. Kesavan, S. Ng, F. de Carvalho, A. Low, and C. Maciocco, "SIP-based IEEE 802.21 media independent handover - a BT Intel collaboration," *BT Technology Journal*, vol. 25, 2007.

[14] Seamless Handover and Security Research [Brief]. Presented by Dr. Subir Das at Applied Communication Sciences, 27 February 2012, in Piscataway, NJ.

[15] Wireshark MIH dissector. Available at https://bugs.wireshark.org/bugzilla/show_bug.cgi?id=5881.

[16] IEEE, *IEEE Standard. 802.11u - 2011*. The Institute of Electrical and Electronic Engineers, Inc., 2011.

Initial Distribution List

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Defense Information Systems Agency
   Ft. Meade, Maryland

3. Dudly Knox Library
   Naval Postgraduate School
   Monterey, California

4. Marine Corps Representative
   Naval Postgraduate School
   Monterey, California

5. Directory, Training and Education, MCCDC, Code C46
   Quantico, Virginia

6. Marine Corps Systems Command
   Quantico, Virginia

7. Marine Corps Tactical System Support Activity (Attn: Operations Officer)
   Camp Pendleton, California